Bedienungsanleitung für

JetProg
Universalprogrammer mit 48 Pin-Treiber, bis 256 Pin-Treiber erweiterungsfähig

BeeProg
Universalprogrammer mit 48 Pin-Treiber, dem USB/LPT - Interface und der IPS-Möglichkeit

LabProg+
Universalprogrammer mit 48 Pin-Treiber

SmartProg
Universalprogrammer mit 40 Pin-Treiber und der IPS-Möglichkeit

PREPROM-02aLV
Universalprogrammer für Speicher

MEMprog
Universalprogrammer für Speicher

—

T51prog
Programmer für Prozessoren der MCS51- und Atmel AVR-Reihen mit ISP-Möglichkeit

51&AVRprog
Programmer für Prozessoren der MCS51- und Atmel AVR-Reihen

PIKprog+
Microchip-PICmicro-Programmer mit ISP-Möglichkeit

PIKprog
Microchip-PICmicro-Programmer

SEEprog
Programmer für serienmäßige EEPROMs

ELNEC s.r.o.
Prešov, Slowakei
Dezember 2003

LabProg+ Best.-Nr. 15 00 53

Programming Times

Note: The programming times of the programmer LabProg+, attached to PC through LPT port, depends a bit on PC speed, LPT port type and operating system free resources. Look at www. elnec.com - ELNEC programmers, tips to increase of programming speed application note for details. If you plan to use high-capacity memories, please accept a long programming time or think about using much more powerful BeeProg Best.-Nr. 15 01 70 or JetProg Best.-Nr. 15 00 21 programmer.

For comparison of programming speed BeeProg/ LabProg+ here is the table:

Device	Operation	Time BeeProg	Time LabProg+
AT29C040A	programming and verify	26 sec.	75 sec.
AM29F040	programming and verify	28 sec.	165 sec.
PIC16C67	programming and verify	6 sec.	30 sec.

COPYRIGHT © 1997 - 2003
ELNEC s.r.o.

Inhalt dieser Bedienungsanleitung ist das geistige Eigentum der Gesellschaft ELNEC s.r.o., Prešov. Sie darf ohne vorherige schriftliche Zustimmung der Gesellschaft ELNEC s.r.o. Prešov weder teilweise noch ganz reproduziert, vervielfältigt, übersetzt oder in irgendeiner Form übertragen werden.

Das Steuerprogramm zu den ELNEC-Programmern ist ebenfalls das geistige Eigentum der Gesellschaft ELNEC s.r.o. Prešov. Das Steuerprogramm und auch kein Teil des Steuerprogramms darf analysiert oder in irgendeiner Form mittels irgendeines Mediums für irgendeinen Zweck modifiziert werden.

Wir haben versucht, die in dieser Bedienungsanleitung angeführten Informationen genau und aktuell zu beschreiben. Die Gesellschaft ELNEC s.r.o. übernimmt keinerlei Haftung für Schäden, die aus falscher Verwendung dieser Bedienungsanleitung resultieren.

Weil jedes Erzeugnis ständiger Entwicklung unterliegt, können bestimmte Verbesserungen vorgenommen werden, die in dieser Bedienungsanleitung nicht enthalten sind. Etwaige Abweichungen von dieser Bedienungsanleitung sind im Internet unter www.elnec.com zu finden.

Die Gesellschaft ELNEC s.r.o., Prešov behält sich das Recht vor, das in dieser Bedienungsanleitung beschriebene Erzeugnis ohne vorherige Warnung jederzeit zu modifizieren und zu vervollkommnen.

In dieser Betriebanweisung wurden Bezeichnungen von Programmprodukten, Firmen usw. benutzt, die durch Schutzrechte oder durch registrierte Schutzrechte jeweiliger Inhaber geschützt werden.

Wie soll diese Bedienungsanleitung benutzt werden?

> ***Sehr geehrter Kunde,***
>
> *herzlichen Dank für Ihr Vertrauen, das Sie uns mit dem Kauf des ELNEC-Programmers entgegengebracht haben.*

Diese Bedienungsanleitung wurde erstellt, um Ihnen zu helfen, unseren Programmer sehr schnell und einfach zu installieren.
Diese Bedienungsanleitung geht davon aus, dass der Anwender über die Grundkenntnisse und Grunderfahrungen hinsichtlich der Bedienung eines PC verfügt und auch mit der Installation der technischen Ausstattung sowie der PC-Software bereits vertraut ist. Ein Teil dieser Bedienungsanleitung, der die Software des Programmers beschreibt, sollte in Verbindung mit der Hilfe-Datei des Steuerprogramms verwendet werden. Diese Hilfe-Datei enthält ebenfalls die Ergänzungen und Zusätze, die erst nach der Herausgabe dieser Bedienungsanleitung zustande gekommen sind.

Hinweis: *Weil dieses Manual für alle ELNEC-Programmer gemeinsam gültig ist, lesen Sie bitte sorgfältig nur die Teile dieser Bedienungsanleitung, die den von Ihnen gekauften Typ des ELNEC-Programmers betreffen.*

Nach dieser Bedienungsanleitung stehen zwei Optionen zur Verfügung:

Schnellstart

Der Schnellstart ist für die erfahrenen Anwender bestimmt und bietet Ihnen nur die Informationen, die Sie für die Installation Ihres Programmers und des Steuerprogramms brauchen werden. Sollten Sie mehr Informationen brauchen, können Sie jederzeit die Option **Ausführliche Beschreibung (Schritt für Schritt)** anwählen. Falls irgendwelche Probleme auftreten, wählen Sie bitte die Option **Wenn Probleme auftreten**.

Ausführliche Beschreibung (Schritt für Schritt)

Schritt für Schritt stellt das bevorzugte Verfahren dar, das insbesondere für die wenig erfahrenen Anwender empfohlen wird, damit sie keine der wichtigen Eigenschaften des ELNEC-Programmers übersehen. „Schritt für Schritt" bedeutet, dass Sie die ausführliche Bedienungsanleitung sorgfältig lesen und aufgrund dessen auch erfahren werden, was alles bei der Hardware- aber auch Software-Installation gemacht werden muss, wobei Sie sich natürlich mit sämtlichen Möglichkeiten bekannt machen können,

die der Programmer aber auch die Software anbietet, um Ihren Programmer voll ausnutzen zu können.

Die aktuelle Version der Bedienungsanleitung finden Sie auf unserer Webseite: www.elnec.com).

Inhalt

Wie soll diese Bedienungsanleitung benutzt werden? 3
Einleitung 9
Lieferumfang der Produkte 13
Die an den Steuerrechner gestellten Anforderungen 14
Schnellstart 15
Ausführliche Beschreibung 17
JetProg 19
Einleitung 20
Beschreibung des JetProg-Programmers 24
Wie wird der JetProg-Programmer am PC angeschlossen 25
Selbsttest und Programmer-Eichung 27
Technische Spezifikation 28
BeeProg 33
Einleitung 34
Beschreibung des BeeProg -Programmers 37
Wie wird der BeeProg -Programmer am PC angeschlossen? 39
In-system Serial Programming mit dem BeeProg-Programmer 41
Selbsttest und Programmer-Eichung 43
Technische Spezifikation 44
LabProg+ 49
Einleitung 50
Beschreibung des LabProg+ -Programmers 53
Wie wird der LabProg+ -Programmer am PC angeschlossen? 54
Selbsttest und Programmer-Eichung 56
Technische Spezifikation 57
SmartProg 61
Einleitung 62
Beschreibung des SmartProg -Programmers 64
Wie wird der SmartProg -Programmer am PC angeschlossen? 65
In-system Serial Programming mit dem SmartProg-Programmer 66
Selbsttest und Programmer-Eichung 68
Technische Spezifikation 69
PREPROM-02aLV 73
Einleitung 74
Beschreibung des PREPROM-02aLV-Programmers 76
Wie wird der PREPROM-02aLV-Programmer am PC angeschlossen? 77
Selbsttest und Programmer-Eichung 79
Technische Spezifikation 80
MEMProg 83
Einleitung 84
Wie wird der MEMProg -Programmer am PC angeschlossen? 86
Selbsttest und Programmer-Eichung 87
Technische Spezifikation 89
T51prog 93
Einleitung 94
Wie wird der T51prog -Programmer am PC angeschlossen? 96
In-system Serial Programming mit dem T51prog-Programmer 97
Selbsttest und Programmer-Eichung 99
Technische Spezifikation 101
51&AVRprog 105
Einleitung 106

Wie wird der 51&AVRProg-Programmer am PC angeschlossen?.... 107
Technische Spezifikation..... 109
PIKprog+ 111
Einleitung..... 112
Wie wird der PIKprog+ -Programmer am PC angeschlossen?..... 114
In-system Serial Programming mit dem PIKprog+-Programmer..... 115
Selbsttest und Programmer-Eichung..... 117
Technische Spezifikation..... 119
PIKprog 123
Einleitung..... 124
Wie wird der PIKprog-Programmer am PC angeschlossen?..... 125
Technische Spezifikation..... 127
SEEprog 129
Einleitung..... 130
Wie wird der SEEprog-Programmer am PC angeschlossen? 131
Technische Spezifikation..... 133
Software 135
Programmer-Software 136
Datei..... 141
Bauteil 145
Puffer..... 159
Optionen..... 165
Diagnostik..... 173
Hilfe 174
Gemeinsame Hinweise 177
Software 178
Hardware..... 179
ISP (In-System Programming)..... 180
Sonstiges..... 184
Wenn Probleme auftreten Gewährleistungs-bedingungen 185
Falls im Zusammenhang mit dem Programmer Probleme auftreten 186
Was muss man tun, wenn festgestellt wird, dass die Bauteile nicht unterstützt werden?..... 189
Gewährleistungsbedingungen 190
Anlage 193
Anlage A - Device Problem Report..... 195
Anlage B - Keep-current-Dienstleistung 197
Anlage C – AlgOR-Dienstleistung..... 199
Anlage D - Registrierkarte 201
Anlage E – CE Zertifikate 203

Die in der Bedienungsanleitung verwendeten Konventionen

Hinweise auf die Funktionen des Steuerprogramms sind im Fettdruck angeführt, z.B. **Lesen, Datei, Bauteil** usw.

Hinweise auf Steuertasten sind mit den Zeichen **< >** gekennzeichnet, z.B. **<F1>**.

Terminologie:

Bauteil Verschiedene Typen von programmierbaren Bauteilen.

ZIF socket Zero insertion force socket = Nullkraft-Sockel

Buffer Ein Teil des Zentralspeichers oder der Festplatte, in dem die Daten gespeichert werden, die für das Bauteilprogrammieren benutzt werden.

Printerport Paralleler PC-Port, der normalerweise zum Anschluss des Druckers benutzt wird.

Hex-Dateiformat

Dateneingabeformat, das mit einem Standard-Texteditor gelesen werden kann, z.B. im Byte 5AH sind als Zeichen die 5 und A eingegeben – das bedeutet 35H und 41H. Eine Zeile eines solchen Dateisatzes enthält ebenfalls die Adresse, an der diese Daten zu finden sind. Der gesamte Dateisatz ist mit einer Prüfsumme gesichert. Das bedeutet, dass zufällige Fehler in einzelnen Daten entdeckt werden können.

Einleitung

In dieser Bedienungsanleitung sind alle ELNEC-Programmer angeführt, d.h.: **JetProg, BeeProg, LabProg+, SmartProg, PREPROM-02aLV, MEMprog, T51prog, 51&AVRprog, PIKprog+, PIKprog** und **SEEprog**.

JetProg ist ein Universalprogrammer und Tester von logischen IO mit 48 Universal-Pin-Treibern in der Grundausstattung, der bis zu 256 Pin-Treiber konfiguriert werden kann. Das bedeutet, dass das Verzeichnis der unterstützten Bauteile ganz einfach um neue Bauteile ergänzt werden kann. Der Programmer stellt technisch gesehen ein Produkt dar, das in der Lage ist, zuverlässiges und schnelles Programmieren zu gewährleisten und das alles zu einem sehr günstigen und konkurrenzfähigen Preis. In seiner Klasse bietet er sicher das beste Preis-Leistungs-Verhältnis.

BeeProg ist ein schneller USB/LPT Universalprogrammer und Tester von logischen IO mit 48 Universal-Pin-Treibern. Das bedeutet, dass das Verzeichnis der unterstützten Bauteile ganz einfach um neue Bauteile ergänzt werden kann. Er hat einen eingebauten ISP-Steckverbinder zum Programmieren von Bauteilen direkt in der Anlage. BeeProg ist ganz universal und vielseitig. BeeProg ist preiswerter Programmer, der im heutigen Markt sicher eins der besten Preis-Leistungs-Verhältnisse bietet.

LabProg+ ist ein Universalprogrammer und Tester von logischen IO mit 48 Universal-Pin-Treibern. Das Verzeichnis der unterstützten Bauteile kann ganz einfach um neue Bauteile ergänzt werden. LabProg+ ist ein professionelles, wirklich universelles Gerät zu einem günstigen Preis, das mit seinem Preis-Leistungs-Verhältnis neue Maßstäbe im Bereich Universalprogrammer setzt.

SmartProg ist ein kleiner, schneller und leistungsfähiger Programmer für alle Typen von programmierbaren Bauteilen. Er hat einen eingebauten ISP-Steckverbinder zum Programmieren von Bauteilen direkt in der Anlage. Er wurde so konzipiert, damit das Verzeichnis der unterstützten Bauteile einfach um neue Bauteile ergänzt werden kann. In seiner Klasse bietet er sicher eins der besten Preis-Leistungs-Verhältnisse.

PREPROM-02aLV ist ein kleiner, aber leistungsstarker Programmer für EPROM/EEPROM/Flash-EPROM/Serial-EEPROM und NVRAM-Tester. Ergänzt um zusätzliche Module ist er in der Lage, EPROM x16 Byte, Mikroprozessoren wie MCS48, MCS51, PICmicro, Atmel, AVR und NEC, GAL/PALCE-Bauteile zu programmieren. Ebenfalls werden von ihm die LV-Bauteile ab 2V unterstützt.

MEMprog ist ein kleiner, tragbarer, extrem schneller und leistungsfähiger Programmer für EPROM, EEPROM, Flash EPROM, NVRAM, serielle EEPROM Speicher und Tester von statischen RAM Speicher.

T51prog ist ein kleiner, leistungsfähiger und extrem schneller Programmer für Monochip-Mikroprozessoren der MCS51- und Atmel AVR-Reihen mit der Möglichkeit des ISP-Programmierens.

Als Bonus programmiert er auch serienmäßige EEPROMs mit IIC-, Microwire- und SPI-Schnittstellen.

51&AVRprog ist ein kleiner, aber leistungsstarker Programmer für Monochip-Mikroprozessoren der MCS51- und Atmel AVR-Reihen. Zusätzlich können mit seiner Hilfe die serienmäßigen EEPROMs mit IIC-, Microwire- und SPI-Schnittstellen programmiert werden.

PIKprog+ ist ein kleiner, leistungsfähiger und extrem schneller Programmer für das Programmieren von Microchip/PICmicro®-Monochip-Mikroprozessoren mit der Möglichkeit des ISP-Programmierens. Ebenfalls kann er zum Programmieren der serienmäßigen EEPROMs mit der IIC-Schnittstelle, Microwire aber auch SPI eingesetzt werden.

PIKprog ist ein kleiner und leistungsfähiger Programmer für das Programmieren der Microchip/PICmicro®-Monochip-Mikroprozessoren. Auch er kann zum Programmieren der serienmäßigen EEPROMs mit IIC-, Microwire-, aber auch SPI-Schnittstellen eingesetzt werden.

SEEprog ist ein Universalprogrammer für serienmäßige EEPROMs in einer 8-Pin-DIL-Packung mit IIC-, SPI-, Microwire-Schnittstellen. Mit seiner Hilfe können z. B. Digitalthermometer programmiert werden. Von dem Programmer können ebenfalls LV-Bauteile (3,3V) programmiert werden.

Alle diese Programmer sind zum Anschluss an IBM-kompatible PCs vorgesehen. Sie können jedoch auch zu PC AT und höheren Typen, von PC Desktop bis zum Palmtop, angeschlossen werden. Man braucht dazu keine spezielle Karte oder Schnittstelle, weil die Programmer an die Computer über einen parallelen Standardport angeschlossen werden, der normalerweise zum Anschluss eines Druckers verwendet wird.

Alle Programmer arbeiten einwandfrei unter den Betriebsystemen DOS, Windows 3.x/95/98/Me/NT/2000/XP.

Das **Steuerprogramm** wird von allen Programmern benutzt (JetProg, BeeProg, LabProg+, SmartProg, PREPROM-02aLV, MEMprog, T51prog, 51&AVRprog, PIKprog+, PIKprog und SEEprog). Das Steuerprogramm ist sehr einfach, seine Bedienung mit Pull-Down-Menü, Funktionstasten und Hilfe-Dateien ist sehr bequem. Zur Verfügung stehen zwei verschiedene Versionen des Steuerprogramms – eins für DOS (PG4U) und eins für WIN95/98/Me/NT/2000/XP (PG4UW).

Hinweis: *Benutzerbetreuung der DOS-Software für unsere Programmer ist seit 7/2003 untergebrochen worden.*

Die Konzeption der Programmer, die Sicherheitsschaltungen und die eingebauten Bauteile haben dazu beigetragen, dass den Kunden, falls sie einen von den MEMprog-, T51prog-, 51&AVRprog-, PIKprog+-, PIKprog- oder SEEprog-Programmern gekauft haben, eine Garantie in Dauer von **1 Jahr** und im Falle der

JetProg-, BeeProg-, LabProg+-, SmartProg-, PREPROM-02aLV-Programmer eine Garantie in Dauer von **3 Jahren** gewährt wird, wobei die Garantie auf 25.000 Zyklen je ein ZIF-Sockel eingeschränkt ist. Diese Garantiebedingungen sind gültig für Kunden, die einen Programmer direkt von der Gesellschaft Elnec kaufen. Garantiebedingungen der Elnec Vertriebsgesellschaften können von der gültigen Staatslegislatur im Land der Vertriebsgesellschaft oder von der Garantiepolitik der Vertriebsgesellschaft abhängig sein.

Im Preis sind enthalten:

- kostenlose Beratungsdienstleistungen (Hotline) im Bereich Programmieren
- Anspruch auf neue kostenlose Software-Versionen

Neue Software-Versionen für das Steuerprogramm stehen Ihnen kostenlos auf unserer Web-Seite: **www.elnec.com** zur Verfügung.

Die Neuigkeit für Unterstützung der Kunden sind die Keep-Current- und AlgOR-Kundendienstleistungen.

- **Keep-Current -** im Rahmen dieser Dienstleistung wird an die Kunden die neueste Version des Steuerprogramms sowie Ergänzungen zu der Bedienungsanleitung (Keep-Current-Paket) übersendet. Die Dienstleistung Keep-Current bietet einfach die höchste Qualität beim Programmieren mit Hilfe der ELNEC-Programmer.
- **AlgOR** (Algorithm On Request – Algorithmen auf Wunsch) Kundendienstleistung, die dem Programmer-Anwender ein Instrument zur Verfügung stellt, aufgrund dessen der Kunde in der Lage ist, die Anzahl sowie das Sortiment von programmierbaren Bauteilen selbst zu wählen sowie zu der Konzeption und zur Verbesserung der Eigenschaften des Steuerprogramms beizutragen.

Anmerkung: *Es wird nicht empfohlen, die LabProg+-, PREPROM-02aLV-, MEMprog-, 51&AVRprog-, PIKprog- und SEEprog-Programmer für das ISP-Programmieren zu verwenden. Dazu siehe die FAQ-Option (Fragen und Antworten) auf unserer Web-Seite* **www.elnec.com**.

Lieferumfang der Produkte

Bevor Sie mit der Installation Ihres Programmers beginnen, überprüfen Sie bitte sorgfältig die Lieferung auf Vollständigkeit: Sollten einige Teile, die in der nachstehenden Liste angeführt sind, fehlen, bzw. beschädigt sein, nehmen Sie bitte umgehend Kontakt mit Ihrem Händler auf.

Lieferumfang für die JetProg-, BeeProg-, LabProg+-, SmartProg, PREPROM-02aLV-, MEMprog-, T51prog- und PIKprog+-Programmer

- Programmer
- Verbindungskabel (mit 2 25-Pin-CANNON-Steckverbindern) für Verbindung des PCs mit dem Programmer
- Netzstromquelle, geeignet für den jeweiligen Programmer
- Diagnostikkopf für Programmer-Selftest
- Staubschutzdeckel für den ZIF-Sockel
- Bedienungsanleitung
- CD mit Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Lieferumfang für die 51&AVRprog-, PIKprog- und SEEprog-Programmer

- Programmer
- Verbindungskabel (mit zwei CANNON-Steckverbindern) für Verbindung des PCs mit dem Programmer
- Netzstromquelle, geeignet für den jeweiligen Programmer
- Bedienungsanleitung
- CD mit dem Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Die an den Steuerrechner gestellten Anforderungen

Mindestanforderungen:

- Prozessor AT 10 MHz (DOS-Software-Version), PC 486 (WIN-Software-Version)
- 512 kB freier RAM-Speicher (DOS-Software-Version), 16MB RAM (WIN-Software-Version)
- 1x CD-Laufwerk
- Festplatte mit 20 MB freiem Speicher
- Betriebssystem DOS 3.2 und höher oder MS Windows 3.xx, WIN95/98/Me/NT/2000/XP
- Freier Parallelport für Druckeranschluss mit nichts verbindet

Hinweis: *Benutzerbetreuung der DOS-Software für unsere Programmer ist seit 7/2003 untergebrochen worden.*

Empfohlene Anforderungen:

- Prozessor Pentium 100 MHz und höher
- 32 MB freier RAM-Speicher
- 1x CD-Laufwerk
- Festplatte mit 20 MB freiem Speicher
- Betriebssystem MS Windows 95/98/Me/NT/2000/XP
- 1x freier bidirektionaler Parallel-Druckerport mit nichts verbindet
- für Jetprog-, BeeProg-, SmartProg-, MEMprog-, T51prog- und PIKprog+-Programmer ein freier IEEE 1284 kompatibel (ECP/EEP) ParallelPort, am PCI-Bus
- für BeeProg-Programmer ein USB 2.0 kompatibel Port

Anmerkung: *Um den Betrieb zu vereinfachen, empfehlen wir Ihnen, eine zusätzliche I/O-Karte mit zusätzlichem LPT-Port einzusetzen. Andernfalls muss zwischen Drucker und Programmer umgesteckt werden.*

Schnellstart

Anschluss des Programmers an einen PC

- Schalten Sie Ihren PC und Programmer aus.
- Verbinden Sie den PC-Parallelport und den Programmer-Kommunikationsport mit dem mitgelieferten Verbindungs-kabel.
- Schalten Sie Ihren PC ein.
- Schließen Sie den Programmer an die Stromquelle an.

Software-Installation

Starten Sie das Installationsprogramm (DOS: Install.exe, WINDOWS: Setup.exe) auf der bereits gelieferten CD-ROM und richten Sie sich dann nach den Anweisungen des Installationsprogramms. Neue Informationen über Hardware und Software, die in dieser Bedienungsanleitung nicht enthalten sind, finden sich auf www.elnec.com.
Hinweis: *Benutzerbetreuung der DOS-Software für unsere Programmer ist seit 7/2003 untergebrochen worden.*

Software

Nach dem Start des PG4U.EXE (DOS)- oder des PG4UW.EXE (WIN)-Programms wird das Anwenderprogramm des Steuerprogramms angezeigt. Das Steuerprogramm wird mit dem Hilfe-Menü und den Befehlstasten gesteuert. Im **Device-Menü** sind Funktionen enthalten, die sich auf die Steuerung des Bauteils beziehen. Im **File-Menü** sind Funktionen enthalten, die sich auf die Bearbeitung der Dateien und Verzeichnisse beziehen. Das **Puffer-Menü** dient zur Manipulation des Puffers.

Bauteilprogrammieren – der kürzeste Weg

Benutzen Sie dazu die Befehlstaste **<Alt+F5>** und unter Eingabe des Namens des Herstellers und der Bezeichnung des Bauteils wählen Sie das Bauteil aus, mit dem Sie arbeiten wollen. Falls Sie den Inhalt eines bereits programmierten Bauteils kopieren wollen, legen Sie diesen bitte in den Sockel des Programmers ein und drücken Sie die **<F7>Taste**. Falls Sie das Bauteil mit Hilfe von Daten auf Festplatte programmieren wollen, drücken Sie die Befehlstaste **<F3>**, wählen Sie das Datenformat, in dem die Dateien auf der Festplatte gespeichert sind und übertragen Sie diese in den Puffer. In den Sockel des Programmers legen Sie das zu programmierende Bauteil ein. Vergewissern Sie sich, ob das Bauteil gelöscht ist - Befehlstaste **<F6>**. Anschließend programmieren Sie das Bauteil - Befehlstaste **<F9>**. Jetzt können Sie den ergänzenden Vergleich des Programmierens vornehmen - Befehlstaste **<F8>**.

Ausführliche Beschreibung

JetProg

Einleitung

JetProg stellt eine neue Generation von ELNEC-Universalprogrammern dar, von denen Windows 95/98/Me/NT/2000/XP unterstützt wird. Der Programmer ist so gestaltet, dass er die auf Entwicklungsabteilungen und Programmierzentren gestellten strengen Anforderungen erfüllen kann.

Vom **JetProg** werden alle Typen und Technologien von programmierbaren Bauteilen unterstützt. Der Programmer stellt ein technisch sehr präzise ausgearbeitetes Produkt dar, dessen Einsatz zuverlässiges und einfaches Programmieren gewährleistet, und das zu einem sehr wettbewerbsfähigen Preis. In seiner Klasse bieter er sicher das beste Preis-Leistungs-Verhältnis.

JetProg ist zum Anschluss an IBM-kompatible PCs vorgesehen. Er kann an PC AT und höhere Typen angeschlossen werden, vom PC Desktop bis zum Palmtop. Am PC wird der Programmer über einen für den Drucker vorgesehenen Parallelport angeschlossen. Beim Anschließen des JetProg-Programmers muss also der PC weder ausgeschaltet noch auseinandergenommen werden. Wir empfehlen, den mit **IEEE 1284 kompatiblen Parallelport (ECP/EEP) am PCI-Bit-Bus** zu verwenden. Das Bedienungsprogramm unterstützt auch die IEEE 1284-Norm.

JetProg programmiert die Bauteile sehr schnell, weil er eine leistungsfähige Hardware auf FPGA-Basis und ECP/EPP (IEEE 1284)-Unterstützung des Parallelports benutzt. Das Programmieren eines AT29C040A dauert etwa 28 Sekunden, es ist also schneller als bei den meisten Mitbewerbern. Man kann sagen, dass der Programmer eine günstige Lösung für Programmierzentren und Einheiten mit einer mittleren Quantität von programmierbaren Bauteilen darstellt.

Standardausführung des JetProg-Programmers:

- JetProg, Grundeinheit
- JetProg, DIL48 Einsockel-Modul

Im nachfolgenden Text versteht man unter „JetProg“ die Grundausführung des JetProg-Programmers.

Die Grundeinheit des **JetProg-Programmers** enthält 48 leistungsfähige Pin-Treiber, die mit Hilfe von "Pin-Treiber Expansion"-Modulen auf insgesamt 256 Pin-Treiber erweitert werden können. Die eingesetzten Hochgeschwindigkeits-bauteile bieten qualitätsgerechte Signale ohne Überschwingungen und können zum Programmieren oder Testen aller Bauteile eingesetzt werden. Der Pin-Treiber liefert Speisespannung, die niedrig genug ist, um alle vorhandenen (1.8V), aber auch zukünftigen Low-Voltage-Bauteile programmieren zu können.

Die Modularkonzeption des Programmers ermöglicht dem Kunden, den Programmer als einen flexiblen Universalprogrammer für Entwicklungsabteilungen zu verwenden oder als Hochleistungsprogrammer in einer Fertigungsstrasse einzusetzen. Das Multiprogrammieren erfolgt über die "Multiple-Socket"-Module.

Der leistungsstarke PIN-Treiber ermöglicht jeden von den 48 PINs ganz unabhängig mit den Werten für H, L, pull-up, pull-down, Takt, Erdung, Speisespannung sowie mit zwei Programmierspannungen zu belegen. Von jedem PIN kann man natürlich auch lesen. Dieser Programmer ermöglicht das Einsetzen von Bauteilen in der DIL-Packung mit 48 PINs, ohne dass ein Adapter oder ein Sondermodul eingesetzt werden muss.

JetProg ist nicht nur Programmer, sondern auch **Tester** für TTL- und CMOS-Schaltbauteile und Speicher. JetProg ermöglicht es, eine bereits vom **Anwender definierte Testsequenz** zu generieren. Er kann also auch als Tester höherer Funktionseinheiten dienen, aber z.B. auch als Kabeltester.

Der Programmer wurde für den **Multitasking-Betrieb** konstruiert. Der Programmer besitzt eigene Intelligenz, die von einem leistungsfähigen Mikroprozessor und von weiteren Bauteilen gebildet wird. Er ist in der Lage, ganz autonom, d.h. unabhängig von dem vorhandenen Zustand des Steuerprogramms in dem PC, jede zeitlich kritische Programmiersequenz vorzunehmen. Aus diesem Grunde läuft die **Arbeit im Hintergrund** unter OS Windows 3.x, Windows 95/98 und Windows NT/2000/XP tatsächlich reibungslos ab.

Mit Hilfe des Programmers kann nicht nur die **richtige Positionierung** (Verschiebung, Drehung) des zu programmierenden Bauteils, sondern auch der Kontakt der einzelnen PINs des zu programmierenden Bauteils im Sockel überprüft werden. Diese Eigenschaften, die durch Limitierung des Speise- aber auch des Programmstromes sowie aufgrund der Überprüfung der Identifizierungs-Bytes des zu programmierenden Bauteils unterstützt werden, gewährleisten, dass seitens Bedienung kein Fehler auftreten kann.

Die **Sicherheitsschaltungen** des Programmers gewährleisten, dass das zu programmierende Bauteil nicht beschädigt wird, falls

Störungen der Speisequelle auftreten oder die Kommunikation mit dem PC unterbrochen wird. In solchen Fällen wird vom Prozessor, der im Programmer - ganz unabhängig von dem PC – eingebaut ist, eine genau spezifizierte Sequenz von einzelnen Schritten durchgeführt und zwar auf die Art, dass nicht nur das zu programmierende Bauteil in Ordnung bleibt, sondern auch die zu programmierende Zelle und der Inhalt des zu programmierenden Bauteils nicht beschädigt wird. Die Hardware des Programmers ermöglicht den **Selbsttest** durchzuführen, d.h. das Steuerprogramm kann jederzeit die Richtigkeit der Pin-Treiber-Funktionen, das Vorhandensein und Richtigkeit aller Spannungen testen, den Prozessor und seine Umgebung, richtige Steuerstufenfolge, aber auch den Kommunikationskanal zwischen dem PC und dem Programmer überprüfen.

Die optimal designte Leiterplatte reduziert während der Arbeit mit dem Bauteil die auf dem Sockel auftretenden negativen Einflüsse auf ein Minimum (Groundbouncing, instabile Stromversorgung). **Sicherheitsschaltungen** auf allen Eingängen des Programmers (Einspeisung- und Kommunikationssteckverbinder, aber auch der Programmiersockel), schützen den Programmer als auch das zu programmierende Bauteil vor einer durch Überspannung oder elektrostatische Entladung hervorgerufenen Beschädigung.

Der **Vergleich** des Programmierens erfolgt durch den Programmer selbst bei **Stromversorgungsgrenzwerten**, die für das jeweilige Bauteil gegeben sind, was natürlich die Richtigkeit des Programmierablaufes erhöht und die Aufbewahrung der in dem Bauteil eingespeicherten Daten über längere Zeit gewährleistet.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, nach Herstellungsbetrieben oder nach vorgeschriebenen Masken gewählt werden, gewährleisten eine sehr einfache und bequeme Orientierung in den unterstützten Bauteilen. Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen) sind um Testfunktionen (illegal bit test, Überprüfung der Lage auf die Richtigkeit) sowie um Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt.

Das Programm gewährleistet komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Datei-formates, und nimmt die erforderliche Datenkonversion vor.

Für den Fall, dass während des Programmierens für jedes Bauteil individuell die Seriennummer eingegeben werden muss, verfügt der JetProg-Programmer über die automatische **Zunahmefunktion**. Diese Funktion gewährleistet, dass jedes Mal, bevor ein neues Bauteil eingesetzt wird, im Puffer die Erhöhung der Seriennummer erfolgt. Die automatische Zunahmefunktion ermöglicht, dass die Seriennummer, bzw. die individuelle Identifikation des zu programmierenden Bauteils ebenfalls aus der jeweiligen Datei gelesen werden kann.

Für den JetProg-Programmer stehen **Sockelkonverter** vom DIL auf PLCC, SOIC, PSOP und TSOP zur Verfügung. Mit Hilfe des Programms wird gesichert, dass z.B. der Konverter DIL44/PLCC44 ermöglicht, mit sämtlichen Bauteilen in der PLCC44-Packung zu arbeiten.
Bauteile, die mit mehr als 48 Pins versehen sind, werden mit Hilfe von:

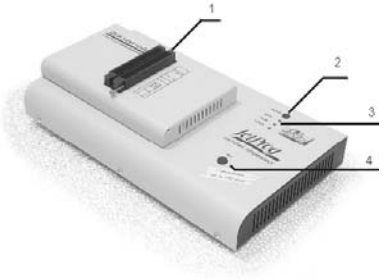
- Pin-Treiber-Expansion-Modulen und von einem Universal-Einsockel-Modul mit entsprechendem Sockel
- von einfachen speziellen Packungs-Konvertern unterstützt.

Der Vorteil der Konstruktion des JetProg-Programmers ist, dass die neuen Bauteile ausschließlich mit einem **Programm unterstützt werden.** Weil die Gesellschaft ELNEC sehr flexibel ist, geschieht es fast sofort. Außerdem sind mit dem Betrieb des Programmers keine zusätzlichen Aufwendungen für neue Software-Versionen, bzw. für aufwendige zusätzliche Software-Module verbunden.

Die Konzeption des JetProg-Programmers, Sicherheitsschaltungen sowie die eingesetzten Bauteile erlauben es, für den Programmer eine **3-jährige Garantieleistung** zu geben, die auf 25.000 Zyklen je ZIF-Sockel eingeschränkt wird.

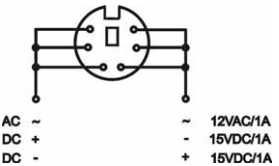
Beschreibung des JetProg-Programmers

① ZIF-Sockel mit 48 PIN-s (**Z**ero **I**nsertion **F**orce)
② LED-Anzeige für Stromversorgung, Anschalten / niedrige Verlustleistung (Sleep – Mode)
③ LED-Anzeige für Arbeit mit dem Bauteil und für Ergebnis dieser Arbeit
④ JA-Taste
⑤ Steckverbindung für Anschluss des Verbindungskabels PC <-> JetProg



⑥ Steckverbindung für Anschluss des Einspeisungskabels
⑦ Steckverbindung für interne Verwendung

Steckverbindung für Anschluss des Einspeisungskabels
(Spannung abhängig von der Version des Programmers)



Anmerkung: *JetProg hat keinen eingebauten Netzschalter, weil es im inaktiven Modus niedrige Leistungsaufnahme hat. Der inaktive Zustand des Programmers wird durch eine kleinere Leuchtintensität der LED „POWER“ angezeigt.*

Wie wird der JetProg-Programmer am PC angeschlossen

Die Sicherheitsschaltungen des Programmers schützen das zu programmierende Bauteil vor einem kurzfristigen oder langfristigen Ausfall der Stromversorgung, teilweise auch in dem Falle, dass an dem PC ein Fehler liegt. Sie verhindern jedoch nicht die eventuelle Vernichtung des zu programmierenden Bauteils bei falsch eingestellten Programmier-Parametern. Das zu programmierende Bauteil sollte z.B. bei einem Reset oder beim Ausschalten des Computers, beim Abkoppeln des Verbindungskabels zum JetProg-Programmer usw. nicht beschädigt werden, es kann aber auch passieren, dass bereits die zu programmierende Zelle beschädigt wird. Ziehen Sie das Bauteil nicht aus dem ZIF-Sockel heraus, wenn es programmiert wird (LED BUSY-Leuchte leuchtet).

Benutzen der LPT-Steckverbindung

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem JetProg-Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel des Druckers heraus und an seiner Stelle schließen Sie das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an einen freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll. Der Austausch des Kabels zum Drucker für den Kabel zum Programmer ist etwas umständlich, es wird nicht jedoch empfohlen, den JetProg-Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Schalter ist nicht möglich!
Sie könnten eine zusätzliche I/O-Karte zu verwenden, damit Sie nicht zwischen Drucker und Programmer umstecken müssen.

Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“ mit voller Intensität; nach und nach blinken alle LED-Leuchten (Selbsttest des Programmers nach dem Einschalten). Sinkt die Leuchtintensität der LED „POWER“, so bedeutet es, der JetProg-Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren.
Starten Sie jetzt das Steuerprogramm für den JetProg-Programmer.

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des JetProg-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

- **Anschluss:** *ZUERST das Kabel zum PC und DANN die Speisung.*
- **Abkopplung:** *ZUERST die Speisung und DANN das Kabel zum PC.*

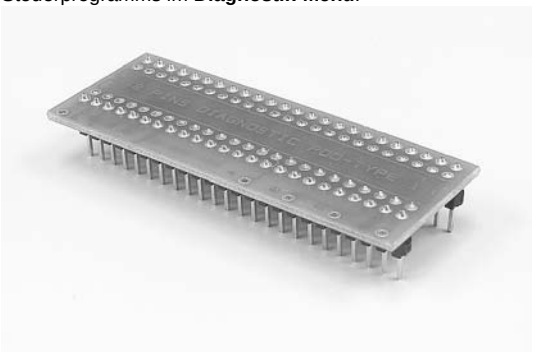
Was den JetProg-Programmer angeht, ist die Reihenfolge beim Anschließen und Abkoppeln der Kabel nicht wichtig – die Sicherheitsschaltungen am Eingang in den Programmer verhindern die Beschädigung der Programmer-Bauteile bei beliebiger Manipulation. **Wir empfehlen jedoch, auch an Ihren PC zu denken…**

Probleme der JetProg ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der JetProg ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

Selbsttest und Programmer-Eichung

Haben Sie das Gefühl, dass Ihr Programmer nicht ganz nach Ihren Erwartungen arbeitet, führen Sie bitte den Selbsttest des Programmers mit Hilfe des Diagnostikkopfes (Diagnostic POD) durch. Der Diagnostikkopf wird serienmäßig zusammen mit dem JetProg-Programmer geliefert. Um während der Arbeit mit dem JetProg-Programmer die optimalen Ergebnisse auf demselben Niveau zu halten, wird es empfohlen, einmal in **sechs** Monaten (falls der Programmer sehr oft eingesetzt wird, einmal in drei Monaten) den Programmer zu testen und die Eichung zu überprüfen. Folgen Sie dabei den Anweisungen des Steuerprogramms im **Diagnostik-Menü**.



Handhabung mit den zu programmierenden Bauteilen

Nachdem Sie das Bauteil gewählt haben, legen Sie dieses in den ZIF-Sockel ein (Hebel in der Stellung oben) und legen Sie den Hebel um, um den Kontakt zwischen dem Bauteil und dem ZIF-Sockel herzustellen.
Die richtige Lage des zu programmierenden Bauteils im ZIF-Sockel ist aus dem Schema neben dem ZIF-Sockel ersichtlich. Das zu programmierende Bauteil soll in den ZIF-Sockel nur in dem Falle eingelegt oder aus dem ZIF-Sockel herausgenommen werden, wenn die LED „BUSY“ nicht mehr leuchtet.

Technische Spezifikation

HARDWARE

Grundeinheit, DAC

- mittels FPGA aufgebauter IEEE1248-Druckerport, Übertragungsgeschwindigkeit bis zum 1MB/s
- leistungfähiger Mikroprozessor (20 MHz) durch FPGA state machine unterstützt, 20MHz getaktet
- drei D/A Konverter für VCCP, VPP1 und VPP2, gesteuerte Flanken- und Abfallflankensteilheit
- VCCP Bereich 0..8V/1A
- VPP1, VPP2 Bereich 0..26V/1A
- Selbstkalibration und Selbsttestfähigkeit
- Einspeisungs- und Kommunikationssteckverbinder, vor Überspannung oder elektrostatischer Entladung geschützt
- Bananenklemme für ESD-Armband

Sockel, Pin-Treiber

- Pin-Treiber: 48 als Standard, max. 256
- 1x VCC, 2x VPP können zu jedem Pin angeschlossen werden
- perfekte GND-Verbindung für jeden Pin
- mittels FPGA aufgebauter TTL-Pin-Treiber ermöglicht alle Pin-Treiber-Pins mit H-Wert, L-Wert, Takt, pull-up und pull-down zu belegen
- Spannungsebenen des Analogteils des Pin-Treibers: Ausgangwert von 1.8V bis zum 26V wählbar
- Strombegrenzung, Überstrom-Unterbrechung, Speisespannungausfall-Unterbrechung
- ESD-Schutz für jeden Sockel-Pin (IEC1000-4-2: 15kV Luft, 8kV Kontakt)
- Test der Leitfähigkeit (ZIF des Programmers – das zu programmierende Bauteil) für jeden Pintreiber-Pin

Sockel, Basis-Konfiguration

- 48-Pin DIL ZIF-Sockel für Bauteile von 8 bis 48 Pins mit Packungsbreite von 300/600 mil

UNTERSTÜTZTE BAUTEILE

Programmer

- EPROM: NMOS/CMOS, 1702*-, 2708*-, 27xxx- und 27Cxxx-Reihen mit einem 8/16-Bit-Bus, einschl. der LV-Versionen
- EEPROM: NMOS/CMOS, 28xxx-, 28Cxxx-, 27EExxx-Reihen mit einem 8/16-Bit-Daten-Bus
- Flash EPROM: 256Kbit bis 32Mbit, 28Fxxx-, 29Cxxx-, 29Fxxx-, 29BVxxx-, 29LVxxx-, 29Wxxx-, 49Fxxx-Reihen, mit einem 8/16-Bit-Daten-Bus, inkl. der LV-Versionen

- Serien-E(E)PROM: 17Xxxx-, 18Vxxx- 24Cxxx-, 24Fxxx-, 25Cxxx-, 45Dxxx-, 59Cxxx-, 85xxx-, 93Cxxx-, EPCxxx-, NVM3060-, MDAxxx-Reihen, inkl. der LV-Versionen
- PROM: AMD, Harris, National, Philips/Signetics, Tesla, TI
- NV RAM: Dallas DSxxx, SGS/Inmos MKxxx, Simtek STKxxx, XICOR 2xxx, ZMD U63x, Benchmarq BQxxx, Catalyst (CSI) CAT22Cxx, Greenwitch Instr. GRxxx, Ramtron FMxxxx, Seiko Instruments S-22xxx, SGS-Thomson (STM), Mxxxxx-Reihen
- PLD: SPLD, CPLD-Reihen von Altera, Atmel, AMD-Vantis, Lattice, NS, Philips, TI, Xilinx, …
- Mikroprozessoren der 48-Reihe: 87x41, 87x42, 87x48, 87x49, 87x50
- Mikroprozessoren der 51-Reihe: 87xx, 87Cxxx, 87LVxx, 89Cxxx, 89Sxxx, 89LVxxx aller Hersteller, Philips 87C748..752, 87LPCxxx
- Mikroprozessoren der Intel 196-Reihe: 87C196 KB/KC/KD/ KT/KR/...
- Mikroprozessoren Atmel AVR: AT90Sxxxx, ATtiny- und ATMega-Reihen
- Mikroprozessoren Microchip PICmicro: PIC12Cxxx, PIC12Fxxx, PIC16C5x, PIC16Cxxx, PIC16Fxxx, PIC17Cxxx , PIC18Cxxx Reihen, einschließlich der LV-Versionen
- Mikroprozessoren Motorola: 68HC05-, 68HC08-, 68HC11-Reihen
- Mikroprozessoren National: COP8xxx-Reihe
- Mikroprozessoren NEC: uPD78Pxxx-Reihe
- Mikroprozessoren Scenix (Ubicom): SX18xxx-, SX20xxx-, SX28xxx-Reihe
- Mikroprozessoren SGS-Thomson: ST6xx-Reihe
- Mikroprozessoren TI: MSP430-Reihe
- Mikroprozessoren Zilog:Z86xx-Reihe
- Mikroprozessoren anderer Firmen: Holtek, Samsung, Scenix, Fujitsu, Hitachi, NEC, Toshiba, Cypress, TI

Anmerkungen:

- *Die Bauteile, die mit * gezeichnet werden, sind veraltet und können nur mit einem zusätzlichen Modul programmiert werden.*
- *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

I.C. Tester

- TTL: 54-,74 S/LS/ALS/H/HC/HCT-Reihen
- CMOS: 4000-, 4500-Reihen
- statische RAM: 6116 .. 624000
- Generierung der Teststufenfolge, die durch den Anwender definiert wird

Unterstützende Packungen

- Unterstützung folgender Packungen: DIP, PLCC, SDIP, SOIC, PSOP, TSOP, TQFP und andere
- DIP-Unterstützung mit Hilfe eines eigenen DIP-Sockels des Programmers mit 48PINs

- Unterstützung der Bauteile in anderen Packungen durch Universaladapter
- Kompatibel mit Adaptern von anderen Herstellern (für die Unterstützung restlicher Packungen, wie z.B. DIP).

Programmiergeschwindigkeit

Anmerkung: *Die Programmierzeiten sind von der PC-Geschwindigkeit, vom Typ des LPT-Ports, von verfügbaren Quellen des Steuersystems, Störfaktoren aus der Umgebung, usw. abhängig. Zum Vergleich werden die Zeiten für zwei PC-Konfigurationen angeführt.*
Diese Zeiten sind nur illustrativ. Aktuelle Zeiten finden Sie auf www.elnec.com (das Obermenü PRODUCTS, das Linksmenü Typ des Programmers).

Bauteil	Operation	Zeit A	Zeit B
27C010	Programmieren und Vergleich	24 sek	20 sek
AT29C040A	Programmieren und Vergleich	38 sek	28 sek
AM29F040	Programmieren und Vergleich	49 sek	42 sek
PIC16C67	Programmieren und Vergleich	8 sek	7 sek
PIC18F452	Programmieren und Vergleich	13 sek	9 sek

Messungsvoraussetzungen:
Zeit A: *Pentium MMX, 250 MHz, ECP/EPP, WIN98.*
Zeit B: *Athlon 750 MHz, ECP/EPP auf PCI-Bit-Bus, WIN98*

SOFTWARE

- **Algorithmen:** nur von Hersteller empfohlene oder zertizifierte Algorithmen sind benutzt.
- **Neue SW-Versionen:** neue Versionen stehen kostenlos zur Verfügung, ungefähr jede 2 Wochen.
- **Haupteigenschaften:** Revision-Historie, Session-Logging, On-Line-Hilfe, Informationen über Bauteile und Algorithmen.

Operationen

- **Standardoperationen:**
  - Intelligente Auswahl des Bauteils nach Typ, Hersteller oder Namenfragments des Bauteils
  - automatische Auswahl der E(E)PROM/Flash EPROMs nach ID-Byte
  - Leertest, Lesen, Vergleich
  - Programmieren
  - Vergleich und Speicherung der Unterschiede auf die Festplatte
  - Löschen
  - Programmieren der Konfigurations- und Sicherheitsbits
  - Illegal bit test
  - Prüfsumme
- **Sicherheitsoperationen:**
  - Einsetzungstest, Umdrehungeinsetzungstest
  - Kontakttest ZIF-Bauteil
  - Identifizierungs-(ID)- Byte-Test
- **Sonderoperationen:**

- voreingestellter Modus (automatischer Programmierstart nach dem Anlegen des Bauteils)
- automatische Zunahme der Seriennummer
- Statistik
- Count-Down-Modus

Puffer-Operationen:

- view/edit, find/replace
- fill/copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Datei laden/speichern

- datei laden/speichern
- keine Ladezeit, weil der Programmer mit dem PC gesteuert ist
- automatische Identifikation des Dateiformats

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, Tektronix, ASCII-SPACE-HEX
- POF (Altera), JEDEC (Ver. 3.0.A), z.B. ABEL, CUPL, PALASM, TANGO PLD, OrCAD PLD, PLD Designer ISDATA, usw.

Die auf den PC gestellten Anforderungen:

*Siehe **Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 12V..15V AC/max. 1A
- Energiebedarf: max. 12W aktiv, etwa 2.5W sleep
- Abmessungen: 275x157x47 mm
- Gewicht: ca. 1,8 kg
- Temperaturbereich: 5°..40°C
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- JetProg-Programmer, Grundeinheit
- JetProg-Programmer, DIL48 Einsockel-Modul
- Verbindungskabel für die Verbindung des PCs mit dem Programmer
- Netzstromquelle 15V DC/1 A
- Diagnostikkopf für den Selbsttest des Programmers
- Staubschutzdeckel für ZIF-Sockel
- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte

- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren
- neue Software-Versionen kostenlos

BeeProg

Einleitung

BeeProg ist das erste Mitglied der neuen Generation von ELNEC-Universalprogrammern, die Windows 95/98/Me/NT/ 2000/XP unterstützen.

Vom **BeeProg** werden alle Typen und Technologien von programmierbaren Bauteilen unterstützt. Der Programmer ist auch mit einem In-circuit-Steckverbinder zum serienmäßigen Programmieren (ISP) ausgestattet, was seinen Gebrauchswert noch steigert.

BeeProg ist nicht nur Programmer, sondern auch **Tester** für TTL- und CMOS-Schaltbauteile und Speicher. BeeProg ermöglicht es, eine bereits vom **Anwender definierte Testsequenz** zu generieren.
BeeProg stellt ein technisch sehr präzise ausgearbeitetes Produkt dar, dessen Einsatz zuverlässiges und einfaches Programmieren gewährleistet, und das zu einem sehr wettbewerbsfähigen Preis. In seiner Klasse bietet er sicher das beste Preis-Leistungs-Verhältnis.

BeeProg programmiert die Bauteile **sehr schnell**, weil er eine leistungsfähige Hardware auf FPGA-Basis benutzt und Hardware des Programmers verarbeitet auch zeitkritische Operationen. Der Programmer ist in seiner Klasse viel schneller, als die meisten Programmer der Mitbewerber.

BeeProg ist zum Anschluss an IBM-kompatible PCs vorgesehen. Er kann an PC AT und höhere Typen angeschlossen werden, vom PC Desktop bis zum Palmtop. Am PC wird der Programmer über einen für den Drucker vorgesehenen Parallelport angeschlossen. Beim Anschließen des BeeProg-Programmers muss also der PC weder ausgeschaltet noch auseinandergenommen werden.

BeeProg hat auf der Basis von FPGA-Bauteilen 48 variabele leistungsstarke TTL-Pintreiber, die es ermöglichen, das Gerät mit den Werten für H, L, pull-up, pull-down und die Lesemöglichkeit für jedes Pin des Sockels zu belegen. Verwendete fortgeschrittene hochwertige und Hochgeschwindigkeitsbauteile bieten ausgezeichnete Signale ohne Überschwingungen oder Ground-Bounce-Effekt zum Programmieren aller unterstützten Bauteile. Der Pin-Treiber bietet eine ausreichend niedrige Speisespannung, um das Programmieren von allen gegenwärtigen (1,8 V) sowie künftigen Low-voltage-Bauteilen zu ermöglichen.

Mit Hilfe des Programmers kann nicht nur die **richtige Positionierung** (Verschiebung, Drehung) des zu programmierenden Bauteils, sondern auch der Kontakt der einzelnen PINs des zu programmierenden Bauteils im Sockel überprüft werden. Diese Eigenschaften, die durch Limitierung des Speise- aber auch des Programmstromes sowie aufgrund der Überprüfung der Identifizierungs-Bytes des zu programmierenden

Bauteils unterstützt werden, gewährleisten, dass seitens Bedienung kein Fehler auftreten kann.

Die eingebauten **Sicherheitsschaltungen** des Programmers gewährleisten, dass der Programmer und/oder ein programmierendes Bauteil nicht beschädigt wird, falls Umfeld oder Operator fehlschlägt. Sicherheitsschaltungen auf allen Eingängen des Programmers (Einspeisung- und Kommunikationssteckverbinder, aber auch der Programmiersockel), schützen den Programmer als auch das zu programmierende Bauteil vor einer durch Überspannung oder elektrostatische Entladung hervorgerufenen Beschädigung bis zu 15kV.

Der Vergleich des Programmierens erfolgt durch den Programmer selbst bei **Stromversorgungsgrenzwerten**, was natürlich die Richtigkeit des Programmierablaufes erhöht und die Aufbewahrung der in dem Bauteil eingespeicherten Daten über längere Zeit gewährleistet.

Zum Programmieren von Bauteilen in anderen als in DIL-Packungen steht ein breites Sortiment an Konvertern von DIL auf PLCC, SOIC und anderen Packungen zur Verfügung.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, Herstellungsbetrieben oder den vorgeschriebenen Masken gewählt werden, gewährleistet eine sehr einfache und bequeme Orientierung in den unterstützten Bauteilen.

Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen) sind um Testfunktionen (illegal bit test, Überprüfung der Lage auf die Richtigkeit) sowie um Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt.

Das Programm gewährleistet komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Datei-formates, und nimmt die erforderliche Datenkonversion vor.

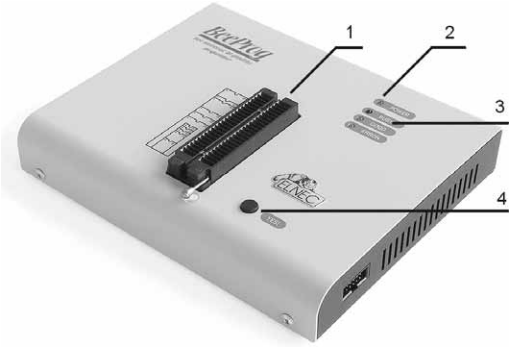
Die **Zunahmefunktion** gewährleistet, dass jedes Mal, bevor ein neues Bauteil eingesetzt wird, im Puffer die Erhöhung der Seriennummer erfolgt. Die automatische Zunahmefunktion ermöglicht, dass die Seriennummer, bzw. die individuelle Identifikation des zu programmierenden Bauteils ebenfalls aus der jeweiligen Datei gelesen werden kann.

Der Vorteil der Konstruktion des BeeProg-Programmers ist, dass die neuen Bauteile ausschließlich mit einem **Programm unterstützt werden.** Weil die Gesellschaft ELNEC sehr flexibel ist, geschieht es fast sofort. Außerdem sind mit dem Betrieb des Programmers keine zusätzlichen Aufwendungen für neue Software-Versionen, bzw. für aufwendige zusätzliche Software-Module verbunden.

Die Konzeption des BeeProg-Programmers, verwendete Bauteile sowie ein zuverlässiger Produktionsprozess erlauben es, für den Programmer eine **3-jährige Garantieleistung** zu geben, die auf 25.000 Zyklen je ZIF-Sockel eingeschränkt wird.

Beschreibung des BeeProg-Programmers

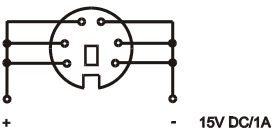
① ZIF-Sockel mit 48 PIN-s (**Z**ero **I**nsertion **F**orce)
② LED-Anzeige für Stromversorgung, Anschalten / niedrige Verlustleistung (Sleep – Mode)
③ LED-Anzeige für Arbeit mit dem Bauteil und für Ergebnis dieser Arbeit
④ JA-Taste



⑤ LPT-Steckverbindung für Anschluss des Verbindungs-kabels PC <-> BeeProg
⑥ USB-Steckverbindung für Anschluss des Verbindungs-kabels PC <-> BeeProg
⑦ Steckverbindung für Anschluss des Einspeisungskabels Steckverbindung für interne Verwendung

⑧ ISP-Steckverbindung

Steckverbindung für Anschluss des Einspeisungskabels



Anmerkung: *BeeProg hat keinen eingebauten Netzschalter, weil es im inaktiven Modus niedrige Leistungsaufnahme hat. Der inaktive Zustand des Programmers wird durch eine kleinere Leuchtintensität der LED „POWER" angezeigt.*

Wie wird der BeeProg-Programmer am PC angeschlossen

Die Sicherheitsschaltungen des Programmers schützen das zu programmierende Bauteil vor einem kurzfristigen oder langfristigen Ausfall der Stromversorgung, teilweise auch in dem Falle, dass an dem PC ein Fehler liegt. Sie verhindern jedoch nicht die eventuelle Vernichtung des zu programmierenden Bauteils bei falsch eingestellten Programmier-Parametern. Das zu programmierende Bauteil sollte z.B. bei einem Reset oder beim Ausschalten des Computers, beim Abkoppeln des Verbindungskabels zum BeeProg-Programmer usw. nicht beschädigt werden, es kann aber auch passieren, dass bereits die zu programmierende Zelle beschädigt wird. Ziehen Sie das Bauteil nicht aus dem ZIF-Sockel heraus, wenn es programmiert wird (LED BUSY-Leuchte leuchtet).

Benutzen der LPT-Steckverbindung

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem BeeProg -Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel des Druckers heraus und an seiner Stelle schließen Sie das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an einen freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll. Der Austausch des Kabels zum Drucker für den Kabel zum Programmer ist etwas umständlich, es wird nicht jedoch empfohlen, den BeeProg -Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Schalter ist nicht möglich!
Sie könnten eine zusätzliche I/O-Karte zu verwenden, damit Sie nicht zwischen Drucker und Programmer umstecken müssen.

Anschalten Sie PC.
Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“ mit voller Intensität; nach und nach blinken alle LED-Leuchten (Selbsttest des Programmers nach dem Einschalten). Sinkt die Leuchtintensität der LED „POWER“, so bedeutet es, der BeeProg-Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren.
Starten Sie jetzt das Steuerprogramm für den BeeProg -Programmer.

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des BeeProg-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

- **Anschluss:** *ZUERST das Kabel zum PC und DANN die Speisung.*
- **Abkopplung:** *ZUERST die Speisung und DANN das Kabel zum PC.*

Was den BeeProg -Programmer angeht, ist die Reihenfolge beim Anschließen und Abkoppeln der Kabel nicht wichtig – die Sicherheitsschaltungen am Eingang in den Programmer verhindern die Beschädigung der Programmer-Bauteile bei beliebiger Manipulation. **Wir empfehlen jedoch, auch an Ihren PC zu denken…**

Benutzen der USB-Steckverbindung

In diesem Fall ist Anschlussreihenfolge des USB-Kabels und des Stromkabels irrelevant.

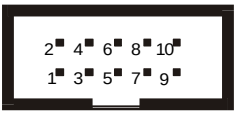
Probleme der BeeProg ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der BeeProg ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

In-system Serial Programming mit dem BeeProg-Programmer

Definitionen, Empfehlungen und Hinweise sind im Teil **Gemeinsame Hinweise/ISP** angeführt.

Beschreibung des ISP-Steckverbinders des BeeProg-Programmers

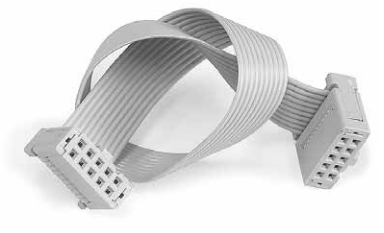


Vorderansicht auf der ISP-Steckverbindung des Programmers.

ISP-Steckerbelegung

Pin	**Bedeutung**
1	VCCP für das Zielbauteil mit Messung
2,4, 6, 8,10	H/L/Lesen, GND, VPP
3	H/L/Lesen, GND, VCCP, VPP
5	Versorgungsspannung für das Zielsystem
7,9	GND

Die konkrete Bedeutung der Pins des ISP-Steckverbinders hängt vom zu programmierenden Bauteil ab und ist im Fenster **Informationen über das Bauteil** dargestellt. Die Bedeutung der Pins entspricht den Empfehlungen aus den Applikationshinweisen, die die Hersteller von Ersatzteilen herausgeben. Verwendete Applikationshinweise können Sie auf www.elnec.com im Teil Applikationshinweise finden (Das Obermenü SUPPORT, das Linksmenü Application notes).
Anmerkung: *Der Pin 1 ist am ISP-Steckverbinder des Kabels durch einen dreieckigen Einschnitt gekennzeichnet.*



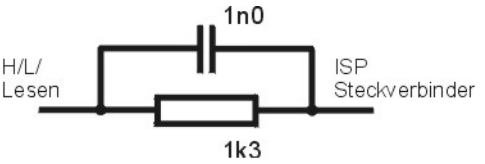
ISP-Kabel des BeeProg-Programmers

Hinweis:

- ***Wenn** Sie den* BeeProg *als **ISP-Programmer** verwenden, **legen Sie** das Bauteil **nicht in den ZIF-Sockel.***
- ***Wenn** Sie Bauteile im **Sockel** programmieren, **stecken Sie das ISP-Kabel nicht in den ISP-Steckverbinder.***
- *Verwenden Sie ausschließlich das **beigelegte ISP-Kabel.** Verwendung eines anderen ISP-Kabels (anderes Material, andere Länge, usw.) kann Grund für ein unzuverlässiges Funktionieren sein.*

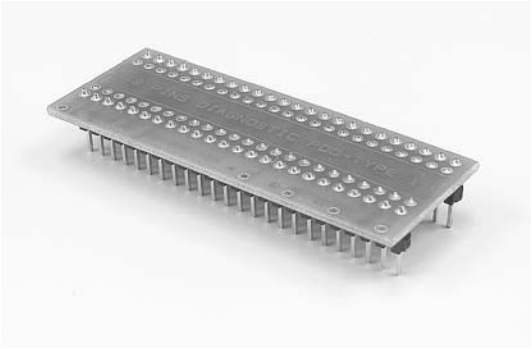
- ***BeeProg** kann nur ein programmierbares Bauteil speisen, aber die Zielanlage **kann nicht den BeeProg speisen.***
- ***BeeProg** führt die Programmierspannung an das Zielbauteil und prüft ihren Wert (das Zielsystem kann die Programmierspannung beeinflussen). Wenn die Programmierspannung von der gewünschten abweicht, wird die Arbeit mit dem Bauteil unterbrochen.*

Anmerkung: *Treiber des BeeProg-Programmers zum H/L/Lesen*



Selbsttest und Programmer-Eichung

Haben Sie das Gefühl, dass Ihr Programmer nicht ganz nach Ihren Erwartungen arbeitet, führen Sie bitte den Selbsttest des Programmers mit Hilfe des Diagnostikkopfes (Diagnostic POD) durch. Der Diagnostikkopf wird serienmäßig zusammen mit dem BeeProg-Programmer geliefert. Um während der Arbeit mit dem BeeProg -Programmer die optimalen Ergebnisse auf demselben Niveau zu halten, wird es empfohlen, einmal in **sechs** Monaten (falls der Programmer sehr oft eingesetzt wird, einmal in drei Monaten) den Programmer zu testen und die Eichung zu überprüfen. Folgen Sie dabei den Anweisungen des Steuerprogramms im **Diagnostik-Menü**.



Technische Spezifikation

HARDWARE

Grundeinheit, DAC

- USB 2.0 kompatibel Port
- mittels FPGA aufgebauter IEEE1248-Druckerport, Übertragungsgeschwindigkeit bis zum 1MB/s
- leistungfähiger Mikroprozessor durch FPGA state machine unterstützt
- drei D/A Konverter für VCCP, VPP1 und VPP2, gesteuerte Flanken- und Abfallflankensteilheit
- VCCP Bereich 0..8V/1A
- VPP1, VPP2 Bereich 0..26V/1A
- Selbstkalibration
- Selbsttestfähigkeit
- Einspeisungs- und Kommunikationssteckverbinder, vor Überspannung oder elektrostatischer Entladung geschützt

Sockel, Pin-Treiber

- DIL ZIF-Sockel mit 48 Pins (Zero Insertion Force) für Bauteile von 8 bis 48 Pins und 300/600 mil
- Pin-Treiber: 48 Universal - Pintreiber
- VCCP / VPP1 / VPP2 können zu jedem Pin angeschlossen werden
- perfekte GND-Verbindung für jeden Pin
- mittels FPGA aufgebauter TTL-Pin-Treiber ermöglicht alle Pin-Treiber-Pins mit H-Wert, L-Wert, Takt, pull-up und pull-down zu belegen
- Spannungsebenen des Analogteils des Pin-Treibers: Ausgangwert von 1.8V bis zum 26V wählbar
- Strombegrenzung, Überstrom-Unterbrechung, Speise-spannungausfall-Unterbrechung
- ESD-Schutz für jeden Sockel-Pin (IEC1000-4-2: 15kV Luft, 8kV Kontakt)
- Test der Leitfähigkeit (ZIF des Programmers – das zu programmierende Bauteil) für jeden Pintreiber-Pin

ISP-Steckverbinder

- 10-Pin male Steckverbinder
- 5 TTL-Treiber unterstützen H (wahlfreier H-Zustand von 1.8V bis 5V), L, CLK, Pull-Up, Pull-Down
- 1x VCCP-Spannung (Spannungsbereich 2V..7V/100mA) und 1x VPP-Spannung (Spannungsbereich 2V..25V/50mA)
- VCCP-Spannung mit Sink/Source-Fähigkeit und Spannung-detection
- Versorgungsspannung für das Zielsystem (Spannungsbereich 2V..6V/250mA)

UNTERSTÜTZTE BAUTEILE

Programmer

- EPROM: NMOS/CMOS, 2708*-, 27xxx- und 27Cxxx-Reihen mit einem 8/16-Bit-Bus, einschl. der LV-Versionen
- EEPROM: NMOS/CMOS, 28xxx-, 28Cxxx-, 27EExxx-Reihen mit einem 8/16-Bit-Daten-Bus
- Flash EPROM: 256Kbit bis 32Mbit, 28Fxxx-, 29Cxxx-, 29Fxxx-, 29BVxxx-, 29LVxxx-, 29Wxxx-, 49Fxxx-Reihen, mit einem 8/16-Bit-Daten-Bus, inkl. der LV-Versionen
- Serien-E(E)PROM: 17Xxxx-, 18Vxxx- 24Cxxx-, 24Fxxx-, 25Cxxx-, 45Dxxx-, 59Cxxx-, 85xxx-, 93Cxxx-, EPCxxx-, NVM3060-, MDAxxx-Reihen, inkl. der LV-Versionen
- PROM: AMD, Harris, National, Philips/Signetics, Tesla, TI
- NV RAM: Dallas DSxxx, SGS/Inmos MKxxx, Simtek STKxxx, XICOR 2xxx, ZMD U63x
- PLD: SPLD, CPLD-Reihen von Altera, Atmel, AMD-Vantis, Lattice, NS, Philips, TI, …
- Mikroprozessoren der 48-Reihe: 87x41, 87x42, 87x48, 87x49, 87x50
- Mikroprozessoren der 51-Reihe: 87xx, 87Cxxx, 87LVxx, 89Cxxx, 89Sxxx, 89LVxxx aller Hersteller, Philips 87C748..752
- Mikroprozessoren der Intel 196-Reihe: 87C196 KB/KC/KD/ KT/KR/...
- Mikroprozessoren Atmel AVR: AT90Sxxxx, ATtiny-Reihen
- Mikroprozessoren Microchip PICmicro: PIC12Cxxx, PIC12Fxxx, PIC16C5x, PIC16Cxxx, PIC16Fxxx, PIC17Cxxx , PIC18Cxxx Reihen
- Mikroprozessoren Motorola: 68HC05-, 68HC08-, 68HC11- Reihen
- Mikroprozessoren National: COP8xxx-Reihe
- Mikroprozessoren NEC: uPD78Pxxx-Reihe
- Mikroprozessoren Scenix (Ubicom): SX18xxx-, SX20xxx-, SX28xxx-Reihe
- Mikroprozessoren SGS-Thomson: ST6xx-Reihe
- Mikroprozessoren TI: MSP430-Reihe
- Mikroprozessoren Zilog:Z86xx-Reihe
- Mikroprozessoren anderer Firmen: Holtek, Samsung, Scenix, Fujitsu, Hitachi, NEC, Toshiba, Cypress, TI

Anmerkungen:

- *Die Bauteile, die mit * gezeichnet werden, sind veraltet und können nur mit einem zusätzlichen Modul programmiert werden.*
- *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

I.C. Tester

- TTL: 54-,74 S/LS/ALS/H/HC/HCT-Reihen
- CMOS: 4000-, 4500-Reihen
- statische RAM: 6116 .. 624000
- Generierung der Teststufenfolge, die durch den Anwender definiert wird

Unterstützende Packungen

- Unterstützung folgender Packungen: DIP, PLCC, SDIP, SOIC, PSOP, TSOP, TQFP und andere
- DIP-Unterstützung mit Hilfe eines eigenen DIP-Sockels des Programmers mit 48PINs
- Unterstützung der Bauteile in anderen Packungen durch Universaladapter
- Kompatibel mit Adaptern von anderen Herstellern (für die Unterstützung restlicher Packungen, wie z.B. DIP).

Programmiergeschwindigkeit

Anmerkung: *Die Programmierzeiten sind von der PC-Geschwindigkeit, vom Typ des LPT-Ports, von verfügbaren Quellen des Steuersystems, Störfaktoren aus der Umgebung, usw. abhängig. Zum Vergleich werden die Zeiten für zwei PC-Konfigurationen angeführt.*
Diese Zeiten sind nur illustrativ. Aktuelle Zeiten finden Sie auf www.elnec.com (das Obermenü PRODUCTS, das Linksmenü Typ des Programmers).

Bauteil	Operation	Zeit A	Zeit B	Zeit C
27C010	Programmieren und Vergleich	24 sek	20 sek	24 sek
AT29C040A	Programmieren und Vergleich	38 sek	28 sek	38 sek
AM29F040	Programmieren und Vergleich	49 sek	42 sek	49 sek
PIC16C67	Programmieren und Vergleich	8 sek	7 sek	8 sek
PIC18F452	Programmieren und Vergleich	13 sek	9 sek	13 sek

Zeit A: *Pentium MMX, 250 MHz, ECP/EPP, WIN98.*
Zeit B: *Athlon, 750 MHz, ECP/EPP on PCI bus, WIN98.*
Zeit C: *Athlon, 750 MHz, USB, WIN98.*

SOFTWARE

- **Algorithmen:** nur von Hersteller empfohlene oder zertizifierte Algorithmen sind benutzt.
- **Neue SW-Versionen:** neue Versionen stehen kostenlos zur Verfügung, ungefähr jede 2 Wochen.
- **Haupteigenschaften:** Revision-Historie, Session-Logging, On-Line-Hilfe, Informationen über Bauteile und Algorithmen.

Bauteilsoperationen

- **Standardoperationen:**
  - Intelligente Auswahl des Bauteils nach Typ, Hersteller oder Namenfragments des Bauteils
  - automatische Auswahl der E(E)PROM/Flash EPROMs nach ID-Byte
  - Leertest, Lesen, Vergleich
  - Programmieren
  - Löschen
  - Programmieren der Konfigurations- und Sicherheitsbits
  - Illegal bit test

- Prüfsumme
- **Sicherheitsoperationen:**
  - Einsetzungstest, Umdrehungeinsetzungstest
  - Kontakttest ZIF-Bauteil
  - Identifizierungs-(ID)- Byte-Test
- **Sonderoperationen:**
  - voreingestellter Modus (automatischer Programmierstart nach dem Anlegen des Bauteils)
  - automatische Zunahme der Seriennummer
  - Statistik
  - Count-Down-Modus

Puffer-Operationen:

- view/edit, find/replace
- fill/copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Datei laden/speichern

- keine Ladezeit, weil der Programmer mit dem PC gesteuert ist
- automatische Identifikation des Dateiformats

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, Tektronix, ASCII-SPACE-HEX
- POF (Altera), JEDEC (Ver. 3.0.A), z.B. ABEL, CUPL, PALASM, TANGO PLD, OrCAD PLD, PLD Designer ISDATA, usw.

Die auf den PC gestellten Anforderungen:

Siehe ***Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 15..18V DC (Gleichstrom) max. 1A
- Energiebedarf: max. 12W aktiv, etwa 2W inaktiv (Sleep – Mode)
- Abmessungen: 160x190x42 [mm] (6.3x7.5x1.7 [inch])
- Gewicht: (ohne externen Adapter) 900g
- Temperaturbereich: 5°C ÷ 40°C (41°F ÷ 104°F)
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- BeeProg-Programmer
- USB-Verbindungskabel, LPT-Verbindungskabel
- ISP-Verbindungskabel
- Diagnostikkopf (POD) für den Selbsttest des Programmers
- Staubschutzdeckel für ZIF-Sockel
- Schaltender Netzstromquelleadapter 100..240V AC/15V DC/1A

- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren (Telefon/ Fax / E-Mail).
- neue Software-Versionen kostenlos (www.elnec.com)

LabProg+

Einleitung

LabProg+ ist ein **Universalprogrammer,** der für alle Bauteil-Typen und Technologiearten der programmierbaren Bauteile eingesetzt werden kann. Der leistungsstarke Pin-Treiber ermöglicht jeden von den 48 Pins ganz unabhängig mit den Werten für H, L, pull-up, pull-down, Takt, Erdung, Speisespannung sowie mit zwei Programmierspannungen zu belegen. Von jedem Pin kann man natürlich auch lesen. Dieser Programmer ermöglicht es, jedes Bauteil in der DIL-Packung mit 48 Pins, ohne dass ein Adapter oder ein Sondermodul eingesetzt werden muss, virtuell zu programmieren, und darüber hinaus alle auf dem Markt zur Zeit oder in Zukunft verfügbaren programmierbaren Bauteile zu unterstützen.

Der LabProg+-Programmer kann nicht nur als Programmer für programmierbare Bauteile, sondern auch als **Tester** für TTL- und CMOS-Schaltbauteile oder auch als Tester von statischen und dynamischen Speichern eingesetzt werden. LabProg+ ermöglicht es, eine bereits von dem **Anwender definierte Testsequenz** zu generieren. d.h. er kann als Tester für höhere Funktionseinheiten, aber auch als Kabeltester eingesetzt werden. LabProg+ ist ein professioneller, wirklich universeller Programmer zu einem günstigen Preis, und mit seinem Preis-Leistungs-Verhältnis ist er für Universalprogrammer wegweisend.

LabProg+-Programmer ist für den Anschluss an IBM-kompatible PCs vorgesehen. Er kann zum PC AT und höheren Typen, vom PC Desktop bis zum Palmtop über einen **parallelen Standardport** angeschlossen werden, der normalerweise zum Anschluss eines Druckers verwendet wird. Von dem Steuerprogramm des Programmers wird ebenfalls der ECP-/EPP-Port unterstützt, was die **Schnelligkeit und Zuverlässigkeit der Datenübertragung** vom PC in den Programmer erhöht.

Der Programmer wurde für den **Multitasking-Betrieb** konstruiert. Der Programmer besitzt eigene Intelligenz, die von einem leistungsfähigen Mikroprozessor und von weiteren Bauteilen gebildet wird. Er ist in der Lage, ganz autonom, d.h. unabhängig von dem vorhandenen Zustand des Steuerprogramms in dem PC, jede zeitlich kritische Programmiersequenz vorzunehmen. Aus diesem Grunde läuft die **Arbeit im Hintergrund** unter Windows 3.x, Windows 95/98/Me/NT/2000/XP tatsächlich reibungslos ab.

Mit Hilfe des Programmers kann nicht nur die **richtige Positionierung** (Verschiebung, Drehung) des zu programmierenden Bauteils, sondern auch der Kontakt der einzelnen PINs des zu programmierenden Bauteils im Sockel überprüft werden. Diese Eigenschaften, die durch Limitierung des Speise- aber auch des Programmstromes sowie aufgrund der Überprüfung der Identifizierungs-Bytes des zu programmierenden Bauteils unterstützt werden, gewährleisten, dass seitens Bedienung kein Fehler auftreten kann.

Die **Sicherheitsschaltung** des Programmers gewährleistet, dass das zu programmierende Bauteil nicht beschädigt wird, falls Störungen der Speisequelle auftreten oder die Kommunikation mit dem PC unterbrochen wird. In solchen Fällen werden vom Prozessor, der in dem Programmer eingebaut ist, genau spezifizierte Schritte ganz unabhängig von dem PC durchgeführt, und zwar auf die Art, dass nicht nur das zu programmierende Bauteil in Ordnung bleibt, sondern auch die zu programmierende Zelle und der Inhalt des zu programmierenden Bauteils nicht beschädigt werden. Die Hardware des Programmers ermöglicht den **Selbsttest** durchzuführen, d.h. das Steuerprogramm kann jederzeit die Richtigkeit der Pin-Treiber-Funktionen, das Vorhandensein und Richtigkeit aller Spannungen testen, den Prozessor und seine Umgebung, richtige Steuerstufenfolge, aber auch den Kommunikationskanal zwischen dem PC und dem Programmer überprüfen.

Die optimal designte Leiterplatte reduziert während der Arbeit mit dem Bauteil die auf dem Sockel auftretenden negativen Einflüsse auf ein Minimum (Groundbouncing, instabile Stromversorgung). **Sicherheitsschaltungen** auf allen Eingängen des Programmers (Einspeise- und Kommunikationssteckverbinder, aber auch der Programmiersockel) schützen den Programmer als auch das zu programmierende Bauteil vor einer durch Überspannung oder elektrostatische Entladung hervorgerufenen Beschädigung.

Der Vergleich des Programmierens erfolgt durch den Programmer selbst bei **Stromversorgungsgrenzwerten**, die für das jeweilige Bauteil gegeben sind, was natürlich die Richtigkeit des Programmierablaufes erhöht und die Aufbewahrung der in dem Bauteil eingespeicherten Daten über längere Zeit gewährleistet.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, Herstellungsbetrieben oder den vorgeschriebenen Masken gewählt werden, gewährleistet eine sehr einfache und bequeme Orientierung in den unterstützten Bauteilen. Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen des Bauteils) sind um Testfunktionen (illegal bit test, Überprüfung der Lage auf die Richtigkeit) sowie um die Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt.

Das Steuerprogramm gewährleistet die komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Dateiformates, und nimmt die erforderliche Datenkonversion vor.

Für den Fall, dass während des Programmierens für jedes Bauteil individuell die Seriennummer eingegeben werden muss, verfügt der LabProg+-Programmer über die automatische **Zunahmefunktion**. Diese Funktion gewährleistet, dass jedes Mal, bevor ein neues Bauteil eingesetzt wird, im Puffer die Erhöhung der Seriennummer erfolgt. Die automatische Zunahmefunktion

ermöglicht, dass die Seriennummer, bzw. die individuelle Identifikation des zu programmierenden Bauteils ebenfalls aus der jeweiligen Datei gelesen werden kann.

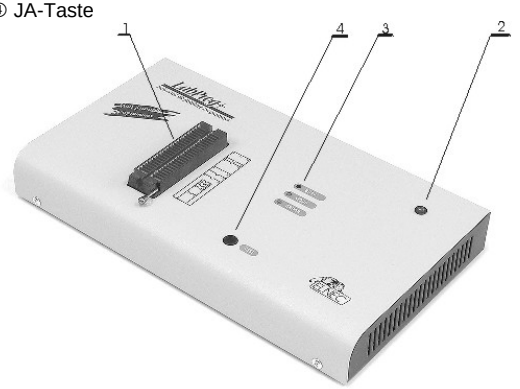
Für den LabProg+ -Programmer stehen **Sockelkonverter** vom DIL auf PLCC, SOIC, PSOP und TSOP zur Verfügung. Mit Hilfe des Programms wird gesichert, dass z.B. ein Konverter DIL44/PLCC44 es ermöglicht, mit sämtlichen Bauteilen in der PLCC44-Packung zu arbeiten.

Der Vorteil der Konstruktion des LabProg+-Programmers ist, dass die neuen Bauteile ausschließlich mit einem **Programm unterstützt werden.** Weil die Gesellschaft ELNEC sehr flexibel ist, geschieht es fast sofort. Außerdem sind mit dem Betrieb des Programmers keine zusätzlichen Aufwendungen für neue Software-Versionen oder zusätzliche Module verbunden.

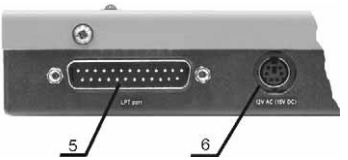
Die Konzeption des LabProg+-Programmers, Sicherheitsschaltungen sowie die eingesetzten Bauteile erlauben es, für den Programmer eine **3-jährige Garantieleistung** zu geben, die auf 25.000 Zyklen je ZIF-Sockel eingeschränkt wird.

Beschreibung des LabProg+ - Programmers

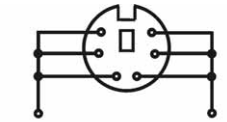
① ZIF-Sockel mit 48 Pins (**Z**ero **I**nsertion **F**orce)
② LED-Anzeige für Stromversorgung
③ LED-Anzeige für Arbeit mit dem Bauteil und für Ergebnis
④ JA-Taste



⑤ Steckverbindung für Anschluss des Verbindungskabels PC <-> LabProg+
⑥ Steckverbindung für Anschluss des Einspeisungskabels



Steckverbindung für Anschluss des Einspeisungskabels (Spannung abhängig von Version des Programmers)



AC	~	~	12VAC/1A
DC	+	-	15VDC/1A
DC	-	+	15VDC/1A

Anmerkung: *LabProg+ hat keinen eingebauten Netzschalter, weil es im inaktiven Modus niedrige Leistungsaufnahme hat. Der inaktive Zustand des Programmers wird durch eine kleinere Leuchtintensität der LED „POWER“ angezeigt.*

Wie wird der LabProg+ - Programmer am PC angeschlossen?

Die Sicherheitsschaltungen des Programmers schützen das zu programmierende Bauteil vor einem kurzfristigen oder langfristigen Ausfall der Stromversorgung, teilweise auch in dem Falle, dass an dem PC ein Fehler liegt. Sie verhindern jedoch nicht die eventuelle Vernichtung des zu programmierenden Bauteils bei falsch eingestellten Programmier-Parametern. Das zu programmierende Bauteil sollte z.B. bei einem Reset oder beim Ausschalten des Computers, beim Abkoppeln des Verbindungskabels zum LabProg+ - Programmer usw. nicht beschädigt werden, es kann aber auch passieren, dass bereits die zu programmierende Zelle beschädigt wird. Ziehen Sie das Bauteil nicht aus dem ZIF-Sockel heraus, wenn es programmiert wird (LED BUSY-Leuchte leuchtet).

Benutzen der LPT-Steckverbindung

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem LabProg+- Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel für den Drucker heraus und an seiner Stelle das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an den freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll. Der Austausch des Kabels zum Drucker für den Kabel zum Programmer ist etwas umständlich, es wird jedoch nicht empfohlen, den LabProg+-Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Druckerumschalter ist nicht möglich! Empfehlenswert ist eine zusätzliche I/O-Karte, auf der sich ein weiterer LPT-Port befindet.

Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“ mit voller Intensität; nach und nach blinken alle LEDs (Selbsttest des Programmers nach dem Einschalten). Sinkt die Leuchtintensität der LED „POWER“, bedeutet es, der LabProg+-Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren.
Jetzt starten Sie das Steuerprogramm für den LabProg+-Programmer.

Anmerkung: *Ist der PC ausgeschaltet und Sie schalten den Programmer ein, kann es passieren, dass die LEDs nicht blinken, weil sich der Programmer im Reset-Modus befinden kann.*

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des LabProg+-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

- **Anschluss:** *ZUERST das Kabel zum PC und DANN die Speisung.*
- **Abkopplung:** *ZUERST die Speisung und DANN das Kabel zum PC.*

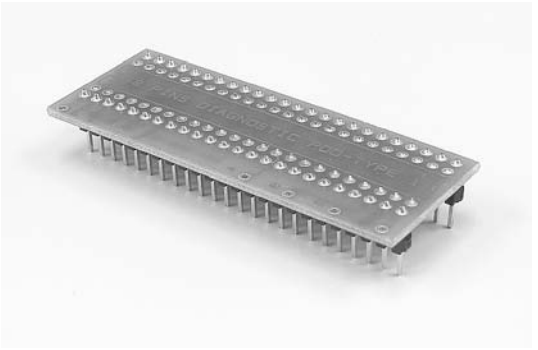
Was den LabProg+-Programmer angeht, ist die Reihenfolge beim Anschließen und Abkoppeln der Kabel nicht wichtig – die Sicherheitsschaltungen am Eingang in den Programmer verhindern die Beschädigung der Programmer-Bauteile bei beliebiger Manipulation. **Wir empfehlen jedoch, auch an Ihren PC zu denken…**

Probleme der LabProg+ ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der LabProg+ ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

Selbsttest und Programmer-Eichung

Haben Sie das Gefühl, dass Ihr Programmer nicht ganz nach Ihren Erwartungen arbeitet, führen Sie bitte den Selbsttest des Programmers mit Hilfe des Diagnostikkopfes (Diagnostic POD) durch. Der Diagnostikkopf wird serienmäßig zusammen mit dem LabProg+-Programmer geliefert. Um während der Arbeit mit dem LabProg+-Programmer die optimalen Ergebnisse auf demselben Niveau zu halten, wird es empfohlen, einmal in **sechs** Monaten (falls der Programmer sehr oft eingesetzt wird, einmal in drei Monaten) den Programmer zu testen und die Eichung zu überprüfen. Folgen Sie dabei den Anweisungen des Steuerprogramms im **Diagnostik-Menü**.



Handhabung mit den zu programmierenden Bauteilen

Nachdem Sie das Bauteil gewählt haben, legen Sie dieses in den ZIF-Sockel ein (Hebel in der Stellung oben) und legen Sie den Hebel um, um den Kontakt zwischen dem Bauteil und dem ZIF-Sockel herzustellen.
Die richtige Lage des zu programmierenden Bauteils im ZIF-Sockel ist aus dem Schema neben dem ZIF-Sockel ersichtlich. Das zu programmierende Bauteil soll in den ZIF-Sockel nur in dem Falle eingelegt oder aus dem ZIF-Sockel herausgenommen werden, wenn die LED „BUSY“ nicht mehr leuchtet.

Technische Spezifikation

Sockel, Pin-Treiber und DAC

- DIL ZIF-Sockel mit 48 Pins (Zero Insertion Force) für Bauteile von 8 bis 48 Pins und 300/600 mil
- 3 D/A Generierungskonverter für das Generieren von VCCP, VPP1 und VPP2, gesteuerte Flanken- und Abfall-flankensteilheit, Stromeinschränkung
- TTL-Treiber unterstützt H, L, CLK, Pull-Up, Pull-Down sowie den dritten Zustand auf allen 48 Pins
- volle Unterstützung der Low-Voltage-Bauteile von 1.8V
- Selbstkalibrierung

Unterstützte Bauteile:

Programmer

- EPROM: NMOS/CMOS, 1702*, 2708*, 27xxx- und 27Cxxx-Reihen von 16Kbit bis 32Mbit mit einem 8/16-Bit-Bus, einschl. der LV-Versionen
- EEPROM: NMOS/CMOS, 28xxx-, 28Cxxx-, 27EExxx-Reihe mit einem 8/16-Bit-Daten-Bus
- Flash EPROM: 256Kbit bis 32Mbit, 28Fxxx-, 29Cxxx-, 29Fxxx-, 29BVxxx-, 29LVxxx-, 29Wxxx-, 49Fxxx-Reihen, mit einem 8/16-Bit-Daten-Bus, inkl. der LV-Versionen
- Serien-E(E)PROM: 17Cxxx-, 18Cxxx, 24Cxxx-, 24Fxxx-, 25Cxxx-, 45Dxxx, 59Cxxx-, 85xxx-, 93Cxxx-, EPCxxx, NVM3060-, MDAxxx-Reihen, inkl. der LV-Versionen
- PROM: Harris, NS, Philips/Signetics, Tesla, TI
- NV RAM: Dallas DSxxx, SGS MKxxx, MKxxx, SIMTEK STKxxx, XICOR2xxx, ZMD U63x, Benchmarq BQxxx, Catalyst (CSI) CAT22Cxx, Greenwitch Instr. GRxxx, Ramtron FMxxxx, Seiko Instruments S-22xxx, SGS-Thomson (STM), Mxxxxx
- PLD: SPLD, CPLD: Altera, Atmel, AMD-Vantis, Lattice, NS, Philips, TI, Xilinx
- Mikroprozessoren der 48-Reihe: 87x41, 87x42, 87x48, 87x49, 87x50
- Mikroprozessoren der 51-Reihe: 87xx, 87Cxxx, 87LVxx, 89Cxxx, 89Sxxx, 89LVxxx, Philips 87C748..752, 87, LPCxxx
- Mikroprozessoren der Intel-Serie 196 KB/KC/KD/KR...,
- Mikroprozessoren Microchip PIC: PIC12Cxxx, PIC12Fxxx, PIC16C5x, PIC16Cxxx, PIC16Fxxx, PIC17Cxxx, PIC18Cxxx
- Mikroprozessoren Atmel AVR: AT90Sxxxx, ATtiny, ATmega
- Mikroprozessoren NEC (uPD78Pxxx-Series), SGS-Thomson (ST6 Series), Zilog (Z8 Series), Motorola (68HC05, 68HC11), NS (COP8), Holtek, Samsung, Scenix, Fujitsu, Hitachi, TI

Anmerkungen:

- *Die Bauteile, die mit * gezeichnet werden, sind veraltet und können nur mit einem zusätzlichen Modul programmiert werden.*

- *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

IC-Tester

- TTL: 54-,74 S/LS/ALS/H/HC/HCT-Reihen
- CMOS: 4000-, 4500-Reihen
- statische RAM: 6116 .. 624000
- Generierung der Teststufenfolge, die durch den Anwender definiert wird

Unterstützende Packungen

- Unterstützung folgender Packungen: DIP, PLCC, SDIP, SOIC, PSOP, TSOP, TQFP
- DIP-Unterstützung mit Hilfe eines eigenen DIP-Sockels des Programmers mit 48Pins
- Unterstützung der Bauteile in anderen Packungen bis 48 Pins durch Universaladapter
- Kompatibel mit Adaptern von anderen Herstellern (für die Unterstützung restlicher Packungen, wie z.B. DIP)

Programmiergeschwindigkeit

Anmerkung:.*Die Programmierzeiten sind von der PC-Geschwindigkeit, vom Typ des LPT-Ports sowie von verfügbaren Quellen des Steuersystems abhängig. Zum Vergleich werden die Zeiten für zwei PC-Konfigurationen angeführt:*
Diese Zeiten sind nur illustrativ. Aktuelle Zeiten finden Sie auf www.elnec.com (das Obermenü PRODUCTS, das Linksmenü Typ des Programmers).

Bauteil	Operation	Zeit A	Zeit B
27C010	Programmieren und Vergleich	39 sek	32 sek
AT29C040A	Programmieren und Vergleich	110 sek	90 sek
AM29F040	Programmieren und Vergleich	165 sek	139 sek
PIC16C67	Programmieren und Vergleich	46 sek	41 sek

Messungsvoraussetzungen:
Zeit A: *Pentium MMX, 250 MHz, ECP/EPP, WIN98.*
Zeit B: *Pentium III, 500 MHz, ECP/EPP, auf PCI-Bit-Bus, WIN98*

Operationen

- **Standardoperationen:**
  - automatische Auswahl der EPROM/EEPROM/Flash EPROMs nach ID-Byte
  - Leertest
  - Lesen
  - Programmieren
  - Vergleich
  - Vergleich und Speicherung der Unterschiede auf die Festplatte
  - Löschen
  - Programmieren der Konfigurations- und Sicherheisbits

- Prüfsumme
- **Sicherheitsoperationen:**
  - Einsetzungstest
  - Kontakttest ZIF-Bauteil
  - Identifizierungs-(ID)- Byte-Test
- **Sonderoperationen:**
  - voreingestellter Modus (automatischer Programmierstart nach dem Einlegen des Bauteils)
  - automatische Zunahme der Seriennummer
  - Statistik
  - Count-Down-Modus

Puffer-Operationen:

- view/edit, find/replace
- fill/copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, Tektronix, ASCII-SPACE-HEX
- POF (Altera), JEDEC (Ver. 3.0.A), z.B. ABEL, CUPL, PALASM, TANGO PLD, OrCAD PLD, PLD Designer ISDATA, usw.

Die auf den PC gestellten Anforderungen:

Siehe ***Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 12V..15V AC/max. 1A
- Energieaufwand: max. 12W aktiv, etwa 1,5W sleep
- Abmessungen: 275x157x47 mm
- Gewicht: ca. 1,5kg
- Temperaturbereich: 5°..40°C,
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- LabProg+-Programmer
- Verbindungskabel für die Verbindung des PCs mit dem Programmer
- Netzstromquelle
- Diagnostikkopf für den Selbsttest des Programmers
- Staubschutzdeckel für ZIF-Sockel
- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren
- neue Software-Versionen kostenlos

SmartProg

Einleitung

SmartProg ist das nächste Mitglied der neuen Generation von ELNEC-Universalprogrammern, die Windows 95/98/Me/NT/2000 /XP unterstützen. Der Programmer wurde entwickelt, um Anforderungen von Entwicklungs- und Servicezentren zu erfüllen, für die die Universalprogrammer der LabProg+-/JetProg-Reihe zu leistungsfähig sind.

Der **SmartProg**-Programmer ist ein kleiner, leistungsstarker und extrem schneller Programmer für alle programmierbaren Bauteile, der zum stationären und mobilen Gebrauch bestimmt ist. Der Programmer arbeitet ohne zusätzliche Module mit allen Bauteilen in DIL-Packung bis zu 40 Pins. Der Programmer ist auch mit einem In-circuit-Steckverbinder zum serienmäßigen Programmieren (ISP) ausgestattet, was seinen Gebrauchswert noch steigert. Der SmartProg ist nicht nur ein Programmer, sondern auch ein RAM-Speicher-Tester.

Dieser Programmer ist ein technisch hervorragend ausgearbeitetes Produkt, das zuverlässiges und schnelles Programmieren zu einem sehr konkurrenzfähigen Preis bietet. In seiner Klasse weist er sicher eins der besten Preis-Leistungs-Verhältnisse vor.

Der Programmer programmiert die Bauteile aus dem Grund sehr schnell, weil die leistungsstarke Hardware auf FPGA-Basis und ECP/EPP(IEEE1284)-Unterstützung des Parallelports verwendet wurde. Der Programmer ist in seiner Klasse viel schneller, als die meisten Programmer der Mitbewerber.

SmartProg-Programmer ist für den Anschluss an alle IBM-kompatible PCs vorgesehen. Er kann zum PC AT und höheren Typen, vom PC Desktop bis zum Palmtop über einen parallelen Standardport angeschlossen werden, der normalerweise zum Anschluss eines Druckers verwendet wird. Beim Anschließen des Programmers muss der PC natürlich weder ausgeschaltet noch auseinandergenommen werden.

SmartProg hat 40 leistungsstarke TTL-Pintreiber, die es ermöglichen, das Gerät mit den Werten für H, L, pull-up, pull-down und die Lesemöglichkeit für jedes Pin des Sockels zu belegen. Verwendete fortgeschrittene hochwertige und Hochgeschwindigkeitsbauteile bieten ausgezeichnete Signale ohne Überschwingungen zum Programmieren aller unterstützten Bauteile. Der Pin-Treiber bietet eine ausreichend niedrige Speisespannung, um das Programmieren von allen gegenwärtigen (1,8 V) sowie künftigen Low-voltage-Bauteilen zu ermöglichen.

Das zu programmierende Bauteil ist vor einer Beschädigung durch Fehler der Bedienung durch leistungsstarke Sicherheitsmaßnahmen geschützt: Kontrolle des **richtigen Einsetzens der Bauteils** in den Sockel, Kontrolle **des richtigen**

Kontaktes der Leitungen, und natürlich die Kontrolle **der elektronischen Identifizierung des Bauteils.**

Die Hardware des Programmers ermöglicht den **Selbsttest** durchzuführen, d.h. das Steuerprogramm kann jederzeit die Richtigkeit der Pin-Treiber-Funktionen, das Vorhandensein und Richtigkeit aller Spannungen testen, den Prozessor und seine Umgebung, richtige Steuerstufenfolge, aber auch den Kommunikationskanal zwischen dem PC und dem Programmer überprüfen.

Der Vergleich des Programmierens erfolgt durch den Programmer selbst bei **Stromversorgungsgrenzwerten**, was natürlich die Richtigkeit des Programmierablaufes erhöht und die Aufbewahrung der in dem Bauteil eingespeicherten Daten über längere Zeit gewährleistet.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, Herstellungsbetrieben oder den vorgeschriebenen Masken gewählt werden, gewährleistet eine sehr einfache und bequeme Orientierung in den unterstützten Bauteilen. Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen des Bauteils) sind um Testfunktionen (illegal bit test, Überprüfung der Lage auf die Richtigkeit) sowie um die Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt.

Das Steuerprogramm gewährleistet die komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Datei-formates, und nimmt die erforderliche Datenkonversion vor.

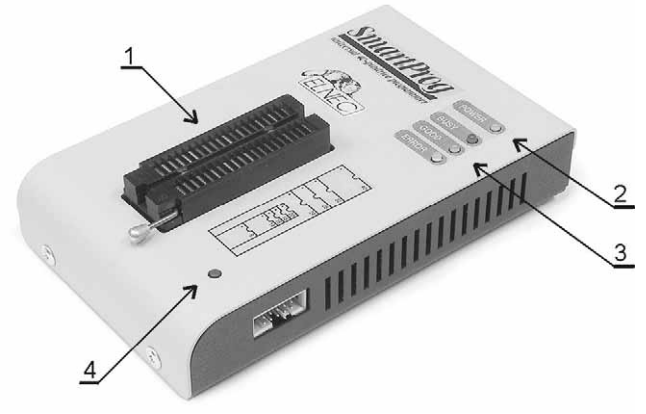
Die **Zunahmefunktion** gewährleistet, dass jedes Mal, bevor ein neues Bauteil eingesetzt wird, im Puffer die Erhöhung der Seriennummer erfolgt. Die automatische Zunahmefunktion ermöglicht, dass die Seriennummer, bzw. die individuelle Identifikation des zu programmierenden Bauteils ebenfalls aus der jeweiligen Datei gelesen werden kann.

Zum Programmieren von Bauteilen in anderen als in DIL-Packungen steht ein breites Sortiment an Konvertern von DIL auf PLCC, SOIC und anderen Packungen zur Verfügung.

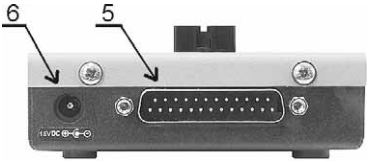
Die Konzeption des SmartProg-Programmers, verwendete Bauteile sowie ein zuverlässiger Produktionsprozess erlauben es, für den Programmer eine **3-jährige Garantieleistung** zu geben, die auf 25.000 Zyklen je ZIF-Sockel eingeschränkt wird.

Beschreibung des SmartProg - Programmers

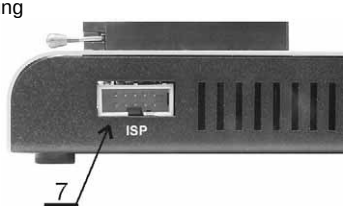
① ZIF-Sockel mit 40 Pins (**Z**ero **I**nsertion **F**orce)
② LED-Anzeige für Stromversorgung
③ LED-Anzeige für Arbeit mit dem Bauteil und das Ergebnis
④ JA-Taste



⑤ Steckverbindung für Anschluss des Verbindungskabels PC <-> SmartProg
⑥ Steckverbindung für Anschluss des Einspeisungskabels



ISP-Steckverbindung



Anmerkung: *SmartProg hat keinen eingebauten Netzschalter, weil es im inaktiven Modus niedrige Leistungsaufnahme hat. Der inaktive Zustand des Programmers wird durch eine kleinere Leuchtintensität der LED „POWER“ angezeigt.*

Wie wird der SmartProg -Programmer am PC angeschlossen?

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem SmartProg- Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel für den Drucker heraus und schließen Sie an seiner Stelle das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an den freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll. Der Austausch des Kabels zum Drucker für das Kabel zum Programmer ist etwas umständlich, es wird jedoch nicht empfohlen, den SmartProg-Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Druckerumschalter ist nicht möglich.

Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“ mit voller Intensität; nach und nach blinken alle LEDs (Selbsttest des Programmers nach dem Einschalten). Sinkt die Leuchtintensität der LED „POWER“, bedeutet es, der SmartProg-Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren.
Jetzt starten Sie das Steuerprogramm für den SmartProg-Programmer.

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des SmartProg-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

- **Anschluss:** *ZUERST das Kabel zum PC und DANN die Speisung.*
- **Abkopplung:** *ZUERST die Speisung und DANN das Kabel zum PC.*

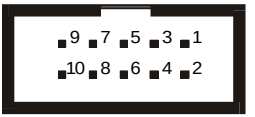
Probleme der SmartProg+ ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der SmartProg ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

In-system Serial Programming mit dem SmartProg-Programmer

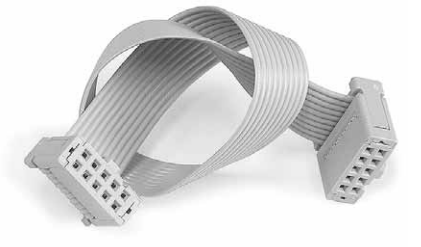
Definitionen, Empfehlungen und Hinweise sind im Teil **Gemeinsame Hinweise/ISP** angeführt.

Beschreibung des ISP-Steckverbinders des SmartProg-Programmers



Pin	Bedeutung
1	VCCP für das Zielbauteil mit Messung
2, 10	H/L/Lesen, GND, VCCP, VPP
3	H/L/Lesen, GND, VPP
4	H/L/Lesen, VPP
5	NC
6	H/L/Lesen, GND, VCCP
7, 9	GND
8	H/L/Lesen, GND

Die konkrete Bedeutung der Pins des ISP-Steckverbinders hängt vom zu programmierenden Bauteil ab und ist im Fenster **Informationen über das Bauteil** dargestellt. Die Bedeutung der Pins entspricht den Empfehlungen aus den Applikationshinweisen, die die Hersteller von Ersatzteilen herausgeben. Verwendete Applikationshinweise können Sie auf www.elnec.com im Teil Applikationshinweise finden (Das Obermenü SUPPORT, das Linksmenü Application notes).
Anmerkung: *Der Pin 1 ist am ISP-Steckverbinder des Kabels durch einen dreieckigen Einschnitt gekennzeichnet.*



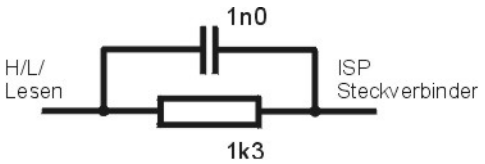
ISP-Kabel des SmartProg-Programmers

Hinweis:

- ***Wenn** Sie den SmartProg als **ISP-Programmer** verwenden, **legen Sie** das Bauteil **nicht in den ZIF-Sockel.***
- ***Wenn** Sie Bauteile im **Sockel** programmieren, **stecken Sie das ISP-Kabel nicht in den ISP-Steckverbinder.***
- *Verwenden Sie ausschließlich das **beigelegte ISP-Kabel.** Verwendung eines anderen ISP-Kabels (anderes Material, andere Länge, usw.) kann Grund für ein unzuverlässiges Funktionieren sein.*

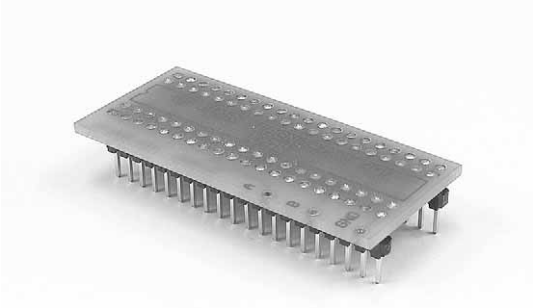
- ***SmartProg** kann nur ein programmierbares Bauteil speisen, aber die Zielanlage **kann nicht den SmartProg speisen.***
- ***SmartProg** führt die Programmierspannung an das Zielbauteil und prüft ihren Wert (das Zielsystem kann die Programmierspannung beeinflussen). Wenn die Programmierspannung von der gewünschten abweicht, wird die Arbeit mit dem Bauteil unterbrochen.*

Anmerkung: *Treiber des SmartProg-Programmers zum H/L/Lesen*



Selbsttest und Programmer-Eichung

Haben Sie das Gefühl, dass Ihr Programmer nicht ganz nach Ihren Erwartungen arbeitet, führen Sie bitte den Selbsttest des Programmers mit Hilfe des Diagnostikkopfes (Diagnostic POD) durch. Der Diagnostikkopf wird serienmäßig zusammen mit dem SmartProg-Programmer geliefert. Um während der Arbeit mit dem SmartProg-Programmer die optimalen Ergebnisse auf demselben Niveau zu halten, wird es empfohlen, einmal in **sechs** Monaten (falls der Programmer sehr oft eingesetzt wird, einmal in drei Monaten) den Programmer zu testen und die Eichung zu überprüfen. Folgen Sie dabei den Anweisungen des Steuerprogramms im **Diagnostik-Menü**.



Handhabung mit den zu programmierenden Bauteilen

Nachdem Sie das Bauteil gewählt haben, legen Sie dieses in den ZIF-Sockel ein (Hebel in der Stellung oben) und legen Sie den Hebel um, um den Kontakt zwischen dem Bauteil und dem ZIF-Sockel herzustellen.
Die richtige Lage des zu programmierenden Bauteils im ZIF-Sockel ist aus dem Schema neben dem ZIF-Sockel ersichtlich. Das zu programmierende Bauteil soll in den ZIF-Sockel nur in dem Falle eingelegt oder aus dem ZIF-Sockel herausgenommen werden, wenn die LED „BUSY“ nicht mehr leuchtet.

Technische Spezifikation

HARDWARE

Programmer

- 2 D/A Generierungskonverter für das Generieren von VCCP, VPP, gesteuerte Flanken- und Abfallflankensteilheit
- VCCP-Umfang: 0V bis 7V/350mA
- VPP-Umfang: 0V bis 25V/200mA
- Mit Hilfe von FPGA realisierter IEEE 1284 slave printer port, Übertragungsgeschwindigkeit bis zu 1MB/s
- Selbstkalibrierung und Selbsttest

Programmiersockel und Pintreiber

- 40-Pin-DIL-Sockel für Bauteile von 8 bis 40 Pins mit der Packungsbreite von 300/600 mil
- Pintreiber: 40 TTL-Pintreiber, Universal-Analogpintreiber (GND/VCC/VPP)
- Der mit Hilfe von FPGA realisierter TTL-Treiber unterstützt H, L, CLK, pull-up, pull-down auf allen Pintreiber-Pins, das H-Niveau von 1,8 V bis 5 V einstellbar
- Spannungen des Analogteils des Pintreibers von 1,8V bis 5V einstellbar
- Steckverbinder für In-circuit-Serial-Programming (ISP)
- Test der Leitfähigkeit (Sockel des Programmers – das zu programmierende Bauteil) für jeden Pintreiber-Pin

Unterstützte Bauteile:

Programmer

- EPROM: NMOS/CMOS, 2708 (*2), 27xxx- und 27Cxxx-Reihen mit einem 8/16-Bit-Bus, einschl. der LV-Versionen (*1)
- EEPROM: NMOS/CMOS, 28xxx-, 28Cxxx-, 27EExxx-Reihen mit einem 8/16-Bit-Daten-Bus(*1)
- Flash EPROM: 28Fxxx-, 29Cxxx-, 29Fxxx-, 29BVxxx-, 29LVxxx-, 29Wxxx-, 49Fxxx-Reihen, mit einem 8/16-Bit-Daten-Bus, inkl. der LV-Versionen (*1*2)
- Serien-E(E)PROM: 17Cxxx-, 24Cxxx-, 24Fxxx-, 25Cxxx-, 59Cxxx-, 85xxx-, 93Cxxx-Reihen, inkl. der LV-Versionen (*1)
- Konfigurations-PROM: 17xxx, inkl. LV-Versionen
- NV RAM: Dallas DSxxx, SGS/Inmos, MKxxx, SIMTEK STKxxx, XICOR2xxx, ZMD U63x
- PLD: von Firmen wie Atmel, AMD-Vantis, Cypress, ICT, Lattice, NS,… (*1)
- Mikroprozessoren der MCS51-Reihe: 87Cxxx, 89Cxxx, 89Sxxx von Atmel, Atmel W&M (Temic), Dallas, LGS, Intel, ISSI, Philips, SST, Winbond (*1*2), Parallel- und Serienmodus (ISP)
- Mikroprozessoren Atmel AVR: ATtiny, AT90Sxxx, Atmega (*1*2), Parallel- und Serienmodus (ISP)

- Mikroprozessoren Microchip PICmicro: PIC12Cxxx, PIC16C5x, PIC16Cxxx, PIC17Cxxx, PIC18Cxxx (*1*2), standardmäßig, auch ISP-Modus
- Mikroprozessoren Scenix (Ubicom): SX18xxx, SX20xxx, SX28xxx

Anmerkungen:

- *(*1) – für non-DIL-Packungen stehen entsprechende Packugskonverter zur Verfügung.*
- *(*2) – für mehr als 40-Pin-Bauteile stehen nur einige Adapter zur Verfügung. Aus diesem Grunde sollten Sie einen leistungsfähigeren Programmer (LabProg+, JetProg) in Betracht ziehen, wenn Sie Bauteile mit mehr als 40 Pins programmieren wollen.*
- *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

IC-Tester

- statische RAM: 6116 .. 624000

Programmiergeschwindigkeit

Anmerkung: *Die Programmierzeiten sind von der PC-Geschwindigkeit, vom Typ des LPT-Ports, von verfügbaren Quellen des Steuersystems sowie von Störfaktoren in der Umgebung abhängig. Zum Vergleich werden die Zeiten für zwei PC-Konfigurationen angeführt:*
Diese Zeiten sind nur illustrativ. Aktuelle Zeiten finden Sie auf www.elnec.com (das Obermenü PRODUCTS, das Linksmenü Typ des Programmers).

Bauteil	Operation	Zeit A	Zeit B
27C010	Programmieren und Vergleich	28 sek	24 sek
AT29C040A	Programmieren und Vergleich	38 sek	28 sek
AM29F040	Programmieren und Vergleich	102 sek	87 sek
PIC16C67	Programmieren und Vergleich	13 sek	11 sek
PIC18F452	Programmieren und Vergleich	11 sek	9 sek
AT89C52	Programmieren und Vergleich	18,5 sek	16,5 sek

Messungsvoraussetzungen:
Zeit A: *Pentium MMX, 250 MHz, ECP/EPP, WIN98.*
Zeit B: *Athlon, 750 MHz, ECP/EPP, auf PCI-Bit-Bus, WIN98*

SOFTWARE

- **Algorithmen:** nur von Hersteller empfohlene oder zertizifierte Algorithmen sind benutzt.
- **Neue SW-Versionen:** neue Versionen stehen kostenlos zur Verfügung, ungefähr jede 2 Wochen.
- **Haupteigenschaften:** Revision-Historie, Session-Logging, On-Line-Hilfe, Informationen über Bauteile und Algorithmen

Operationen

- **Standardoperationen:**
  - Intelligente Auswahl des Bauteils nach Typ, Hersteller oder Namenfragments des Bauteils
  - Leertest, Lesen, Vergleich
  - Programmieren
  - Vergleich und Speicherung der Unterschiede auf die Festplatte
  - Löschen
  - Programmieren der Konfigurations- und Sicherheisbits
  - Illegal bit test
  - Prüfsumme
- **Sicherheitsoperationen:**
  - Einsetzungstest
  - Kontakttest ZIF-Bauteil
  - Identifizierungs-(ID)-Byte-Test
- **Sonderoperationen:**
  - automatische Zunahme der Seriennummer
  - Statistik
  - Count-Down-Modus

Puffer-Operationen:

- view/edit, find/replace
- fill, copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Datei laden/speichern

- keine Ladezeit, weil der Programmer mit dem PC gesteuert ist
- automatische Identifikation des Dateiformats

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, Tektronix, ASCII-SPACE-HEX

Die auf den PC gestellten Anforderungen:

*Siehe **Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 15V DC, max. 0,5A
- Energieaufwand: max. 6W aktiv, etwa 1,4W sleep
- Abmessungen: 165x95x35 mm
- Gewicht: ca. 0,5 kg (ohne Netzstromquelle)
- Temperaturbereich: 5°..40°C
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- SmartProg-Programmer
- Verbindungskabel für die Verbindung des PCs mit dem Programmer
- ISP-Kabel
- Netzstromquelle 220VAC/15VDC, 500 mA
- Diagnostikkopf für den Selbsttest des Programmers
- Staubschutzdeckel für ZIF-Sockel
- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren
- neue Software-Versionen kostenlos

PREPROM-02aLV

Einleitung

PREPROM-02aLV ist ein kleiner und leistungsfähiger **Programmer** für das Programmieren von EPROM-, EEPROM-Flash-EPROM- sowie Serien-EEPROM-Speichern, der aber auch als **Tester** von statischen RAMs eingesetzt werden kann. Der Programmer ist für das professionelle Programmieren bestimmt, stationär oder mobil. Ergänzt um zusätzliche Module ist er in der Lage, ebenfalls EPROM 1Kx8, EPROM xxxK x16 Bits, Mikroprozessoren der 48- und der 51-Reihen, Mikroprozessoren Microchip PIC, Atmel AVR und NEC, GAL/PALCE- und NV RAM-Bauteile zu programmieren. Ein frei programmierbarer Speisespannungs- und Programmspannungs-Generator, aber auch die Einschränkung des logischen H-Niveaus gewährleisten die Low-Voltage-Unterstützung (LV) der Bauteile mit einer Spannung von 2V. Der PREPROM-02aLV-Programmer ist mit seiner Leistung, mit seinen Abmessungen, aber auch dank Vielseitigkeit nicht nur für den Bereich Außendienst, sondern auch in kleinen Betrieben einsetzbar. Aufgrund der Erfahrungen mit den ELNEC-Programmern wird für den PREPROM-02aLV-Programmer eine dreijährige Garantie durch den Hersteller gewährt.

PREPROM-02aLV-Programmer ist zum Anschluss an einen IBM-komplatiblen PC vorgesehen. Er kann jedoch auch zu PC AT und höheren Typen, von PC Desktop bis zum Palmtop, angeschlossen werden. Der Programmer wird am Computer über einen parallelen Standardport angeschlossen, der normalerweise zum Anschluss eines Druckers verwendet wird. Beim Anschließen des PREPROM-02aLV-Programmers an einen PC ist es also nicht notwendig, den PC auszuschalten oder auseinanderzunehmen.

Die Sicherheitsschaltung des Programmers gewährleistet, dass das zu programmierende Bauteil nicht beschädigt wird, falls Störungen der Speisequelle auftreten oder die Kommunikation mit dem PC unterbrochen wird. Die Hardware des Programmers ermöglicht es, einen **Selbsttest** durchzuführen, d.h. das Steuerprogramm kann jederzeit die Richtigkeit der Pin-Treiber-Funktionen, das Vorhandensein und Richtigkeit aller Spannungen zu testen, den Prozessor und seine Umgebung, richtige Steuerstufenfolge, aber auch den Kommunikationskanal zwischen dem PC und dem Programmer zu überprüfen.

Der Vergleich des Programmierens erfolgt durch den Programmer selbst bei Stromversorgungsgrenzwerten, die für das jeweilige Bauteil gegeben sind, was natürlich die Richtigkeit des Programmierablaufes erhöht und die Aufbewahrung der in dem Bauteil eingespeicherten Daten über längere Zeit gewährleistet.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Die Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, nach Herstellungsbetrieben oder nach den vorgeschriebenen Masken gewählt werden, gewährleistet eine sehr einfache und bequeme Orientierung in den unterstützten

Bauteilen. Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen) sind um Testfunktionen (illegal bit test, Überprüfung der Lage auf die Richtigkeit) sowie um die Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt. Das Programm gewährleistet eine komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Dateiformates.

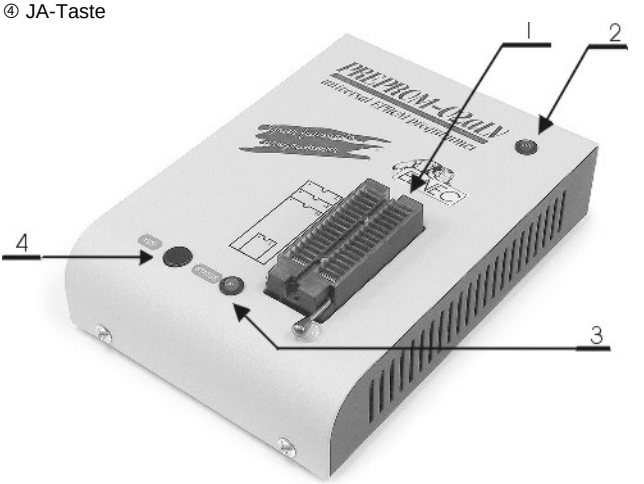
Für den Fall, dass während des Programmierens für jedes Bauteil individuell die Seriennummer eingegeben werden muss, verfügt der Programmer über die automatische **Zunahmefunktion**. Diese Funktion gewährleistet, dass jedes Mal, bevor ein neues Bauteil eingesetzt wird, im Puffer die Erhöhung der Seriennummer erfolgt. Die automatische Zunahmefunktion ermöglicht, dass die Seriennummer, bzw. die individuelle Identifikation des zu programmierenden Bauteils ebenfalls aus der jeweiligen Datei gelesen werden kann.

Für den PREPROM-02aLV-Programmer stehen **Sockelkonverter** vom DIL auf PLCC, SOIC und PSOP zur Verfügung.

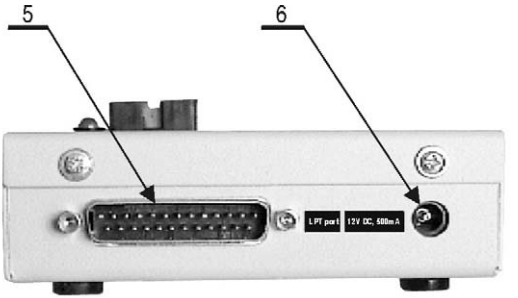
Die Konzeption des PREPROM-02aLV-Programmers, die Sicherheitsschaltungen sowie die eingebauten Bauteile ermöglichen, für diesen Programmer eine dreijährige Garantie zu gewähren, die auf 25.000 Zyklen je ZIF-Sockel eingeschränkt ist.

Beschreibung des PREPROM-02aLV-Programmers

① ZIF-Sockel mit 32 Pins (**Z**ero **I**nsertion **F**orce)
② LED-Anzeige für Stromversorgung
③ LED-Anzeige für Arbeit mit dem Bauteil und für Ergebnis dieser Arbeit
④ JA-Taste



⑤ Steckverbindung für Anschluss des Verbindungskabels PC <-> PREPROM-02aLVProg
⑥ Steckverbindung für Anschluss des Einspeisungskabels



Anmerkung: Der *PREPROM-02aLV-Programmer wird für Dauerbetrieb ausgelegt und braucht aus diesem Grunde nach der Beendigung der Arbeit nicht unbedingt ausgeschaltet zu werden.*

Wie wird der PREPROM-02aLV-Programmer am PC angeschlossen?

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem PREPROM-02aLV-Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel für den Drucker heraus und an seiner Stelle schließen Sie das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an den freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll.
Es wird nicht empfohlen, den PREPROM-02aLV-Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Druckerumschalter ist nicht möglich.

Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“, der PREPROM-02aLV-Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren.
Jetzt starten Sie das Steuerprogramm für den PREPROM-02aLV-Programmer.

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des PREPROM-02aLV-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

- **Anschluss:** *ZUERST das Kabel zum PC und DANN die Speisung.*
- **Abkopplung:** *ZUERST die Speisung und DANN das Kabel zum PC.*

Probleme der PREPROM-02aLV ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der PREPROM-02aLV ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

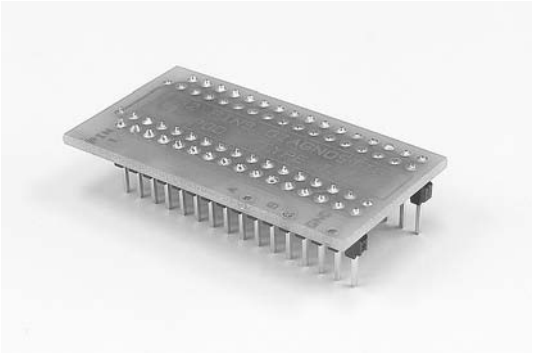
Hinweis für die Laptop- und PREPROM-02aLV-Besitzer

Weil der PREPROM-02aLV-Programmer mit der Nennspannung 12V gespeist wird, möchten wir darauf hinweisen, dass der PREPROM-02aLV-Programmer in Verbindung mit einem Notebook oder Laptop überall dort betrieben werden kann, wo diese

Spannung vorhanden ist, z.B. im Auto. Falls notwendig, kann der Programmer mit zehn NiCd-Akkus im vollen Einsatz länger als eine Stunde betrieben werden. Die Umpolung hat keinen Einfluss, denn der Programmer ist gegen die Umpolung der Speisespannung geschützt. Polarität eines Verbindungskabels, das Sie selbst anfertigen können:

Selbsttest und Programmer-Eichung

Haben Sie das Gefühl, dass Ihr Programmer nicht ganz nach Ihrer Erwartungen arbeitet, führen Sie bitte den Selbsttest des Programmers mit Hilfe des Diagnostikkopfes (Diagnostic POD) durch. Der Diagnostikkopf wird serienmäßig zusammen mit dem PREPROM-02aLV-Programmer geliefert. Um während der Arbeit mit dem PREPROM-02aLV-Programmer die optimalen Ergebnisse auf dem selben Niveau zu halten, wird es empfohlen, einmal in **sechs** Monaten (falls der Programmer sehr oft eingesetzt wird, einmal in drei Monaten) den Programmer zu testen und die Eichung zu überprüfen. Folgen Sie dabei den Anweidungen des Steuerprogramms im **Diagnostik-Menü**.



Handhabung mit den zu programmierenden Bauteilen

Nachdem Sie das Bauteil gewählt haben, legen Sie dieses in den ZIF-Sockel ein (Hebel in der Stellung oben) und legen Sie den Hebel um, um den Kontakt zwischen dem Bauteil und dem ZIF-Sockel herzustellen.
Die richtige Lage des zu programmierenden Bauteils im ZIF-Sockel ist aus dem Schema neben dem ZIF-Sockel ersichtlich. Das zu programmierende Bauteil soll in den ZIF-Sockel nur in dem Falle eingelegt oder aus dem ZIF-Sockel herausgenommen werden, wenn die LED „BUSY“ nicht mehr leuchtet.

Technische Spezifikation

Sockel, Pin-Treiber und DAC

- DIL ZIF-Sockel mit 32 Pins (Zero Insertion Force) für Bauteile von 8 bis 32 Pins mit Packungsbreite von 300/600 mil
- 2 D/A Generierungskonverter für das Generieren von VCCP und VPP, gesteuerte Flanken- und Abfallflankensteilheit, Stromeinschränkung
- TTL-Treiber unterstützt H, L sowie Lesen auf allen 32 Pins
- volle Unterstützung der Low–Voltage-Bauteile von 2V

Unterstützte Bauteile

Programmer

- EPROM: NMOS/CMOS, 27xxx- und 27Cxxx-Reihen, mit einem 8/16*-Bit-Bus
- EEPROM: NMOS/CMOS, 28xxx-, 28Cxxx-, 27Eexxx-Reihen mit einem 8/16*-Bit-Daten-Bus
- Flash EPROM: 256Kbit bis 32Mbit, 28Fxxx-, 29Cxxx-, 29Fxxx-, 29BVxxx-, 29LVxxx-, 29Wxxx-, 49Fxxx-Reihen, mit einem 8/16*-Bit-Daten-Bus
- NonVolatile-RAM: Dallas DSxxx, SGS MKxxx
- Serien-E(E)PROM: 17Cxxx*-, 24Cxxx-, 24Fxxx-, 25Cxxx-, 59Cxxx-, 85xxx-Reihen
- PLD*: AMD PALCE, GALy, PEELy
- Mikroprozessoren der 48-Reihe*: 87x41, 87x42, 87x48, 87x49, 87x50
- Mikroprozessoren der 51-Reihe*: 87xx, 87Cxxx, 89Cxxx, 89Sxxx, Philips 87C748..752
- Mikroprozessoren Microchip PIC*: PIC12Cxxx, PIC16C5x, PIC16Cxxx, PIC18Cxxx
- Mikroprozessoren Atmel AVR*: AT90Sxxxx
- Mikroprozessoren NEC* : uPD78Pxxx

Anmerkungen:

- *Die mit dem * Zeichen gekennzeichneten Bauteile werden mit einem zusätzlichen Modul programmiert. Konkrete Typen der zusätzlichen Module sind in der Liste der unterstützten Bauteile (Datei PG2LVDEV.TXT) zu finden.*
- *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

IC-Tester

- statische RAMs: 6116 .. 624000

Programmiergeschwindigkeit

- M27C256, Programmieren und Vergleich, PC486/DX50: 16 s
- M27C010, Programmieren und Vergleich, PC586/100: 40 s

Operationen

- **Standardoperationen:**
  - automatische Auswahl der EPROM/EEPROM/Flash EPROMs nach ID-Byte
  - Leertest
  - Lesen
  - Programmieren
  - Vergleich
  - Vergleich und Speicherung der Unterschiede auf die Festplatte
  - Löschen
  - Programmieren der Konfigurations- und Sicherheisbits
- **Sicherheitsoperationen:**
  - Identifizierungs-(ID)- Byte-Test
- **Sonderoperationen:**
  - automatische Zunahme der Seriennummer

Puffer-Operationen:

- view/edit, find/replace
- fill/copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, Tektronix, ASCII-SPACE-HEX
- JEDEC

Die auf den PC gestellten Anforderungen:

*Siehe **Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 12V...20V DC/max. 500mA
- Energieaufwand: max. 6W
- Abmessungen: 160x110x50 mm
- Gewicht: ca. 650g
- Umgebung: Büro
- Temperaturbereich: 5°..40°C
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- PREPROM-02aLV-Programmer
- Verbindungskabel für die Verbindung des PCs mit dem Programmer
- Netzstromquelle: 220VAC/12VDC, 500 mA
- Diagnostikkopf für den Selbsttest des Programmers
- Staubschutzdeckel für ZIF-Sockel
- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren
- neue Software-Versionen kostenlos

MEMprog

Einleitung

MEMprog ist das nächste Mitglied der Fachprogrammer von ELNEC-Universalprogrammern, die Windows 95/98/Me/NT/2000 /XP unterstützen. Der Programmer wurde entwickelt, um Anforderungen von Entwicklungs- und Servicezentren zu erfüllen, für die die Universalprogrammer zu leistungsfähig sind.

MEMprog ist ein kleiner, tragbarer, extrem schneller und leistungsfähiger Programmer für EPROM, EEPROM, Flash EPROM, NVRAM, serielle EEPROM Speicher (einschließlich Low-voltage-Bauteilen) und Tester von statischen RAM Speicher.

Der **MEMprog**-Programmer stellt technisch gesehen ein Produkt dar, das in der Lage ist, zuverlässiges und schnelles Programmieren zu gewährleisten und das alles zu einem sehr günstigen und konkurrenzfähigen Preis. In seiner Klasse bietet er sicher das beste Preis-Leistungs-Verhältnis.

Der Programmer programmiert die Bauteile aus dem Grund sehr schnell, weil die leistungsstarke Hardware auf FPGA-Basis und ECP/EPP(IEEE1284)-Unterstützung des Parallelports verwendet wurde.

MEMprog-Programmer ist für den Anschluss an alle IBM-kompatible PCs vorgesehen. Er kann zum PC AT und höheren Typen, vom PC Desktop bis zum Palmtop über einen parallelen Standardport oder ECP/EPP(IEEE1284) kompatiblen Port angeschlossen werden, der normalerweise zum Anschluss eines Druckers verwendet wird.

MEMprog hat 40 leistungsstarke TTL-Pintreiber, die es ermöglichen, das Gerät mit den Werten für H, L, pull-up, pull-down und die Lesemöglichkeit für jedes Pin des Sockels zu belegen. Verwendete fortgeschrittene hochwertige und Hochgeschwindigkeitsbauteile bieten ausgezeichnete Signale ohne Überschwingungen zum Programmieren aller unterstützten Bauteile. Der Pin-Treiber bietet eine ausreichend niedrige Speisespannung, um das Programmieren von allen gegenwärtigen (1,8 V) sowie künftigen Low-voltage-Bauteilen zu ermöglichen.

Das zu programmierende Bauteil ist vor einer Beschädigung durch Fehler der Bedienung durch leistungsstarke Sicherheitsmaß-nahmen geschützt: Kontrolle des **richtigen Einsetzens der Bauteils** in den Sockel, Kontrolle **des richtigen Kontaktes der Leitungen,** und natürlich die Kontrolle **der elektronischen Identifizierung des Bauteils.**

Die Hardware des Programmers ermöglicht den **Selbsttest** durchzuführen, d.h. das Steuerprogramm kann jederzeit die Richtigkeit der Pin-Treiber-Funktionen, das Vorhandensein und Richtigkeit aller Spannungen testen, den Prozessor und seine Umgebung, richtige Steuerstufenfolge, aber auch den

Kommunikationskanal zwischen dem PC und dem Programmer überprüfen.

Der Vergleich des Programmierens erfolgt durch den Programmer selbst bei **Stromversorgungsgrenzwerten**, was natürlich die Richtigkeit des Programmierablaufes erhöht und die Aufbewahrung der in dem Bauteil eingespeicherten Daten über längere Zeit gewährleistet.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, Herstellungsbetrieben oder den vorgeschriebenen Masken gewählt werden, gewährleistet eine sehr einfache und bequeme Orientierung in den unterstützten Bauteilen.

Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen des Bauteils) sind um Testfunktionen (illegal bit test, Überprüfung der Lage auf die Richtigkeit) sowie um die Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt.

Das Steuerprogramm gewährleistet die komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Datei-formates, und nimmt die erforderliche Datenkonversion vor.

Die **Zunahmefunktion** gewährleistet, dass jedes Mal, bevor ein neues Bauteil eingesetzt wird, im Puffer die Erhöhung der Seriennummer erfolgt. Die automatische Zunahmefunktion ermöglicht, dass die Seriennummer, bzw. die individuelle Identifikation des zu programmierenden Bauteils ebenfalls aus der jeweiligen Datei gelesen werden kann.

Zum Programmieren von Bauteilen in anderen als in DIL-Packungen steht ein breites Sortiment an Konvertern von DIL auf PLCC, SOIC und anderen Packungen zur Verfügung.

Wie wird der MEMprog -Programmer am PC angeschlossen?

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem MEMprog-Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel für den Drucker heraus und schließen Sie an seiner Stelle das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an den freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll. Der Austausch des Kabels zum Drucker für den Kabel zum Programmer ist etwas umständlich, es wird jedoch nicht empfohlen, den MEMprog-Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Druckerumschalter ist nicht möglich.

Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“ mit voller Intensität; nach und nach blinken alle LEDs (Selbsttest des Programmers nach dem Einschalten). Sinkt die Leuchtintensität der LED „POWER“, bedeutet es, der MEMprog-Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren. Jetzt starten Sie das Steuerprogramm für den MEMprog-Programmer.

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des MEMprog-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

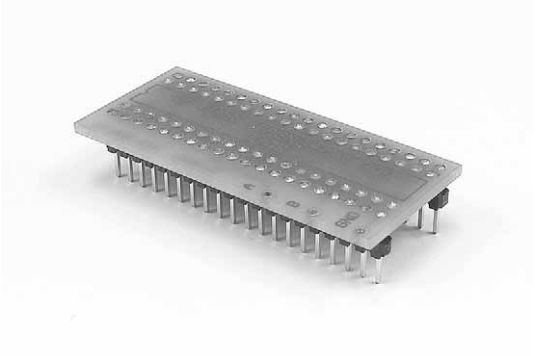
- **Anschluss:** *ZUERST das Kabel zum PC und DANN die Speisung.*
- **Abkopplung:** *ZUERST die Speisung und DANN das Kabel zum PC.*

Probleme der MEMprog ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der MEMprog ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

Selbsttest und Programmer-Eichung

Haben Sie das Gefühl, dass Ihr Programmer nicht ganz nach Ihren Erwartungen arbeitet, führen Sie bitte den Selbsttest des Programmers mit Hilfe des Diagnostikkopfes (Diagnostic POD) durch. Der Diagnostikkopf wird serienmäßig zusammen mit dem MEMprog-Programmer geliefert. Um während der Arbeit mit dem MEMprog-Programmer die optimalen Ergebnisse auf demselben Niveau zu halten, wird es empfohlen, einmal in **sechs** Monaten (falls der Programmer sehr oft eingesetzt wird, einmal in drei Monaten) den Programmer zu testen und die Eichung zu überprüfen. Folgen Sie dabei den Anweisungen des Steuerprogramms im **Diagnostik-Menü**.



Handhabung mit den zu programmierenden Bauteilen

Nachdem Sie das Bauteil gewählt haben, legen Sie dieses in den ZIF-Sockel ein (Hebel in der Stellung oben) und legen Sie den Hebel um, um den Kontakt zwischen dem Bauteil und dem ZIF-Sockel herzustellen.
Die richtige Lage des zu programmierenden Bauteils im ZIF-Sockel ist aus dem Schema neben dem ZIF-Sockel ersichtlich. Das zu programmierende Bauteil soll in den ZIF-Sockel nur in dem Falle eingelegt oder aus dem ZIF-Sockel herausgenommen werden, wenn die LED „BUSY“ nicht mehr leuchtet.

Warnung*: Der MEMprog-Programmer ist mit keinen Sicherheitsschaltungen ausgestattet, die den Inhalt des zu programmierenden Bauteils in kritischen Zuständen schützen würden (z.B. Stromausfall, Unterbrechung der Kommunikation mit dem PC). Das Bauteil kann ebenfalls in dem Falle zerstört werden, wenn während des Programmierens der Ablauf des Steuerprogramms plötzlich unterbrochen wird (RESET, Computer ausschalten...) oder wenn das Bauteil aus dem ZIF-Sockel herausgenommen wird. Durch falsches Einlegen des Bauteils in*

den ZIF-Sockel kann das zu programmierende Bauteil entweder beschädigt oder zerstört werden.

Technische Spezifikation

HARDWARE

Programmer

- 2 D/A Generierungskonverter für das Generieren von VCCP, VPP, gesteuerte Flanken- und Abfallflankensteilheit
- VCCP-Umfang: 0V bis 7V/350mA
- VPP-Umfang: 0V bis 25V/200mA
- Mit Hilfe von FPGA realisierter IEEE 1284 slave printer port, Übertragungsgeschwindigkeit bis zu 1MB/s
- Selbstkalibrierung
- Selbsttest

Programmiersockel (ZIF) und Pintreiber

- 40-Pin-DIL-Sockel für Bauteile von 8 bis 40 Pins mit der Packungsbreite von 300/600 mil
- Pintreiber: 40 TTL-Pintreiber, Universal-Analogpintreiber (GND/VCC/VPP)
- Der mit Hilfe von FPGA realisierter TTL-Treiber unterstützt H, L, CLK, pull-up, pull-down auf allen Pintreiber-Pins, das H-Niveau von 1,8 V bis 5 V einstellbar
- Test der Leitfähigkeit (Sockel des Programmers – das zu programmierende Bauteil) für jeden Pintreiber-Pin

Unterstützte Bauteile:

Programmer

- EPROM: NMOS/CMOS, 2708 (*3), 27xxx- und 27Cxxx-Reihen mit einem 8/16-Bit-Bus, einschl. der LV-Versionen (*1*2)
- EEPROM: NMOS/CMOS, 28xxx-, 28Cxxx-, 27EExxx-Reihen mit einem 8/16-Bit-Daten-Bus(*1*2)
- Flash EPROM: 28Fxxx-, 29Cxxx-, 29Fxxx-, 29BVxxx-, 29LVxxx-, 29Wxxx-, 49Fxxx-Reihen, mit einem 8/16-Bit-Daten-Bus, inkl. der LV-Versionen (*1*2)
- Serien-E(E)PROM: AT17Cxxx-, 24Cxxx-, 24Fxxx-, 25Cxxx-, 59Cxxx-, 85xxx-, 93Cxxx-, NVM3060, MDA206X-Reihen, inkl. der LV-Versionen (*1)
- Konfigurations-PROM: 17xxx-Reihe, inkl. LV-Versionen (*1)
- NV RAM: Dallas DSxxx-, SGS/Inmos MKxxx-, SIMTEK STKxxx-, XICOR 2xxx-, ZMD U63x- Reihen

Anmerkungen:

- *(*1) – für non-DIL-Packungen stehen entsprechende Packugskonverter zur Verfügung.*
- *(*2) – für mehr als 40-Pin-Bauteile stehen nur einige Adapter zur Verfügung. Aus diesem Grunde sollten Sie einen leistungsfähigeren Programmer (LabProg+, JetProg) in Betracht*

ziehen, wenn Sie Bauteile mit mehr als 40 Pins programmieren wollen.

- *(*3) – diese Bauteile werden mit einem zusätzlichen Modul programmiert.*
- *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

IC-Tester

- statische RAM: 6116 .. 624000

Programmiergeschwindigkeit

Anmerkung: *Die Programmierzeiten sind von der PC-Geschwindigkeit, vom Typ des LPT-Ports, von verfügbaren Quellen des Steuersystems sowie von Störfaktoren in der Umgebung abhängig. Zum Vergleich werden die Zeiten für zwei PC-Konfigurationen angeführt:*
Diese Zeiten sind nur illustrativ. Aktuelle Zeiten finden Sie auf www.elnec.com (das Obermenü PRODUCTS, das Linksmenü Typ des Programmers).

Bauteil	**Operation**	**Zeit** A	**Zeit** B
27C010	Programmieren und Vergleich	28 sec	24 sec
AT29C040A	Programmieren und Vergleich	38 sec	28 sec
AM29F040	Programmieren und Vergleich	102 sec	87 sec

Zeit A: *Pentium MMX, 250 MHz, ECP/EPP, WIN98.*
Zeit B: *Athlon, 750 MHz, ECP/EPP on PCI bus, WIN98.*

SOFTWARE

- **Algorithmen:** nur von Hersteller empfohlene oder zertizifierte Algorithmen sind benutzt.
- **Neue SW-Versionen:** neue Versionen stehen kostenlos zur Verfügung, ungefähr jede 2 Wochen.
- **Haupteigenschaften:** Revision-Historie, Session-Logging, On-Line-Hilfe, Informationen über Bauteile und Algorithmen

Bauteiloperationen

- **Standardoperationen:**
  - Intelligente Auswahl des Bauteils nach Typ, Hersteller oder Namenfragments des Bauteils
  - Leertest, Lesen, Vergleich
  - Programmieren
  - Löschen
  - Illegal bit test
  - Prüfsumme
- **Sicherheitsoperationen:**
  - Einsetzungstest
  - Kontakttest ZIF-Bauteil
  - Identifizierungs-(ID)-Byte-Test
- **Sonderoperationen:**

- automatische Zunahme der Seriennummer
- Statistik
- Count-Down-Modus

Puffer-Operationen:

- view/edit, find/replace
- fill, copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Datei laden/speichern

- keine Ladezeit, weil der Programmer mit dem PC gesteuert ist
- automatische Identifikation des Dateiformats

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, Tektronix, ASCII-SPACE-HEX

Die auf den PC gestellten Anforderungen:

Siehe ***Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 12..15V DC, max. 500mA
- Energieaufwand: max. 6W aktiv
- Abmessungen: 132x66x40 [mm] (5.2x2.6x1.6 [inch])
- Gewicht: (ohne externen Adapter) ca. 200g
- Temperaturbereich: 5°C ÷ 40°C (41°F ÷ 104°F)
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- MEMprog-Programmer
- LPT-Verbindungskabel
- Diagnostikkopf (POD) für den Selbsttest des Programmers
- Staubschutzdeckel für ZIF-Sockel
- Netzstromquelleadapter 12V DC/500mA, nicht stabilisiert
- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren (Telefon/ Fax / E-Mail).

- neue Software-Versionen kostenlos (www.elnec.com)

T51prog

Einleitung

T51prog stellt die neue Generation von spezialisierten ELNEC-Programmern, die Windows 95/98/Me/NT/2000/XP unterstützen. Der Programmer wurde in Zusammenarbeit und im Auftrag der Firma Atmel W&M entwickelt, um alle Anforderungen ans Programmieren von existierenden und geplanten Mikroprozessoren der MCS-51-Reihe der Firma Atmel W&M zu erfüllen.

Der **T51prog**-Programmer ist ein kleiner, leistungsstarker und extrem schneller Serviceprogrammer für alle Monochip-Mikroprozessoren der MCS51- und Atmel AVR-Reihen (bis zu 40 Pins DIL), der zum stationären und mobilen Gebrauch bestimmt ist. Der Programmer programmiert auch Serien-EEPROMs mit IIC-, Microwire- und SPI-Schnittstellen. Der Programmer ist auch mit einem **In-circuit-Steckverbinder zum serienmäßigen Programmieren** (ISP) ausgestattet, was seinen Gebrauchswert noch steigert. Der T51prog ist für all jene bestimmt, die mit Mikroprozessoren der MCS51-Familie, bzw. Atmel-AVR-Mikroprozessoren arbeiten, und für die ein Universalprogrammer ungeeignet ist, sei es aus Preisgründen oder aus Gründen der Mobilität und Abmessungen.

Der Programmer ist ein technisch hervorragend ausgearbeitetes Produkt, das zuverlässiges und schnelles Programmieren zu einem sehr wettbewerbsfähigen Preis bietet. In seiner Klasse weist er sicher eins der besten Preis-Leistungs-Verhältnisse vor.

Der Programmer programmiert die Bauteile aus dem Grund sehr schnell, weil die leistungsstarke Hardware auf FPGA-Basis und ECP/EPP(IEEE1284)-Unterstützung des Parallelports verwendet wurde. Der Programmer ist in seiner Klasse viel schneller, als die meisten Programmer der Mitbewerber.

Der **T51prog**-Programmer ist für den Anschluss an alle IBM-kompatible PCs vorgesehen. Er kann zum PC AT und höheren Typen, vom PC Desktop bis zum Palmtop über einen parallelen Standardport angeschlossen werden, der normalerweise zum Anschluss eines Druckers verwendet wird. Beim Anschließen des Programmers muss der PC natürlich weder ausgeschaltet noch auseinandergenommen werden.

Der **T51prog** hat 40 leistungsstarke TTL-Pin-Treiber, die es ermöglichen, das Gerät mit den Werten für H, L, pull-up, pull-down und die Lesemöglichkeit für jeden Pin des Sockels zu belegen. Verwendete fortgeschrittene hochwertige und Hochgeschwindigkeitsbauteile bieten ausgezeichnete Signale ohne Überschwingungen zum Programmieren aller unterstützten Bauteile. Der Pin-Treiber bietet eine ausreichend niedrige Speisespannung, um das Programmieren von allen gegenwärtigen (1,8 V) sowie künftigen Low-voltage-Bauteilen zu ermöglichen.

Mit Hilfe des Programmers kann nicht nur die **richtige Positionierung** (Verschiebung, Drehung) des zu programmierenden Bauteils, sondern auch der Kontakt der einzelnen PINs des zu programmierenden Bauteils im Sockel überprüft werden.

Die Hardware des Programmers ermöglicht den **Selbsttest** durchzuführen, d.h. das Steuerprogramm kann jederzeit die Richtigkeit der Pin-Treiber-Funktionen, das Vorhandensein und Richtigkeit aller Spannungen testen, den Prozessor und seine Umgebung, richtige Steuerstufenfolge, aber auch den Kommunikationskanal zwischen dem PC und dem Programmer überprüfen.

Der Vergleich des Programmierens erfolgt durch den Programmer selbst bei **Stromversorgungsgrenzwerten**, was natürlich die Richtigkeit des Programmierablaufes erhöht und die Aufbewahrung der in dem Bauteil eingespeicherten Daten über längere Zeit gewährleistet.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, Herstellungsbetrieben oder den vorgeschriebenen Masken gewählt werden, gewährleistet eine sehr einfache und bequeme Orientierung in den unterstützten Bauteilen. Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen des Bauteils) sind um Testfunktionen (illegal bit test, Überprüfung der Lage auf die Richtigkeit) sowie um die Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt.

Das Programm gewährleistet die komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Dateiformats, und nimmt die erforderliche Datenkonversion vor.

Die **Zunahmefunktion** gewährleistet, dass jedes Mal, bevor ein neues Bauteil eingesetzt wird, im Puffer die Erhöhung der Seriennummer erfolgt. Die automatische Zunahmefunktion ermöglicht, dass die Seriennummer, bzw. die individuelle Identifikation des zu programmierenden Bauteils ebenfalls aus der jeweiligen Datei gelesen werden kann.

Für den T51prog-Programmer stehen **Sockelkonverter** vom DIL auf PLCC und SOIC zur Verfügung.

Wie wird der T51prog -Programmer am PC angeschlossen?

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem T51prog-Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel für den Drucker heraus und schließen Sie an seiner Stelle das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an den freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll. Der Austausch des Kabels zum Drucker für den Kabel zum Programmer ist etwas umständlich, es wird jedoch nicht empfohlen, den T51prog-Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Druckerumschalter ist nicht möglich.

Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“ mit voller Intensität; nach und nach blinken alle LEDs (Selbsttest des Programmers nach dem Einschalten). Sinkt die Leuchtintensität der LED „POWER“, bedeutet es, der T51prog-Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren. Jetzt starten Sie das Steuerprogramm für den T51prog-Programmer.

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des T51prog-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

- **Anschluss:** *ZUERST das Kabel zum PC und DANN die Speisung.*
- **Abkopplung:** *ZUERST die Speisung und DANN das Kabel zum PC.*

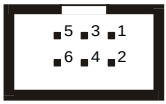
Probleme der T51prog ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der T51prog ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

In-system Serial Programming mit dem T51prog-Programmer

Definitionen, Empfehlungen und Hinweise sind im Teil **Gemeinsame Hinweise/ISP** angeführt.

Beschreibung des ISP-Steckverbinders des T51prog-Programmers



Pin	Bedeutung
1, 3, 4, 5	H/L/Lesen
2	VCC-Messung in der Zielanlage
6	GND

Die konkrete Bedeutung der Pins des ISP-Steckverbinders hängt vom zu programmierenden Bauteil ab und ist im Fenster **Informationen über das Bauteil** dargestellt. Die Bedeutung der Pins entspricht den Empfehlungen aus den Applikationshinweisen der Firma Atmel AVR910 In-System Programming. Diese Applikationshinweise können Sie auf www.elnec.com im Teil Applikationshinweise finden (Das Obermenü SUPPORT, das Linksmenü Application notes).

Anmerkung: *Der Pin 1 ist am ISP-Steckverbinder des Kabels durch einen dreieckigen Einschnitt gekennzeichnet.*

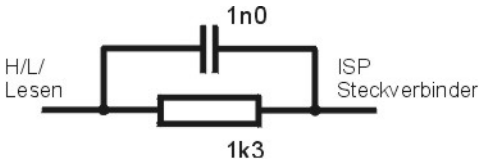
ISP-Kabel des T51prog-Programmers

Hinweis:

- **Wenn** Sie den T51prog als **ISP-Programmer** verwenden, **legen Sie** das Bauteil **nicht in den ZIF-Sockel.**
- **Wenn** Sie Bauteile im **Sockel** programmieren, **stecken Sie das ISP-Kabel nicht in den ISP-Steckverbinder.**
- Verwenden Sie ausschließlich das **beigelegte ISP-Kabel.** Verwendung eines anderen ISP-Kabels (anderes Material, andere Länge, usw.) kann Grund für ein unzuverlässiges Funktionieren sein.

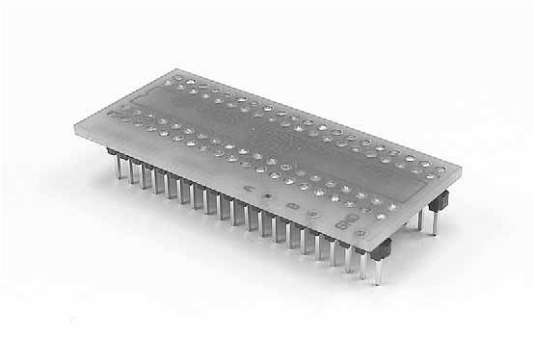
- **T51prog** kann nur ein programmierbares Bauteil speisen, aber die Zielanlage **kann nicht den T51prog speisen.**

Anmerkung: *Treiber des T51prog-Programmers zum H/L/Lesen.*



Selbsttest und Programmer-Eichung

Haben Sie das Gefühl, dass Ihr Programmer nicht ganz nach Ihren Erwartungen arbeitet, führen Sie bitte den Selbsttest des Programmers mit Hilfe des Diagnostikkopfes (Diagnostic POD) durch. Der Diagnostikkopf wird serienmäßig zusammen mit dem T51prog-Programmer geliefert. Um während der Arbeit mit dem T51prog-Programmer die optimalen Ergebnisse auf demselben Niveau zu halten, wird es empfohlen, einmal in **sechs** Monaten (falls der Programmer sehr oft eingesetzt wird, einmal in drei Monaten) den Programmer zu testen und die Eichung zu überprüfen. Folgen Sie dabei den Anweisungen des Steuerprogramms im **Diagnostik-Menü**.



Handhabung mit den zu programmierenden Bauteilen

Nachdem Sie das Bauteil gewählt haben, legen Sie dieses in den ZIF-Sockel ein (Hebel in der Stellung oben) und legen Sie den Hebel um, um den Kontakt zwischen dem Bauteil und dem ZIF-Sockel herzustellen.
Die richtige Lage des zu programmierenden Bauteils im ZIF-Sockel ist aus dem Schema neben dem ZIF-Sockel ersichtlich. Das zu programmierende Bauteil soll in den ZIF-Sockel nur in dem Falle eingelegt oder aus dem ZIF-Sockel herausgenommen werden, wenn die LED „BUSY“ nicht mehr leuchtet.

Warnung*: Der T51prog-Programmer ist mit keinen Sicherheitsschaltungen ausgestattet, die den Inhalt des zu programmierenden Bauteils in kritischen Zuständen schützen würden (z.B. Stromausfall, Unterbrechung der Kommunikation mit dem PC). Das Bauteil kann ebenfalls in dem Falle zerstört werden, wenn während des Programmierens der Ablauf des Steuerprogramms plötzlich unterbrochen wird (RESET, Computer ausschalten...) oder wenn das Bauteil aus dem ZIF-Sockel herausgenommen wird. Durch falsches Einlegen des Bauteils in*

den ZIF-Sockel kann das zu programmierende Bauteil entweder beschädigt oder zerstört werden.

Technische Spezifikation

Sockel, Pin-Treiber und DAC

- mit Hilfe von FPGA realisierter IEEE 1284 slave printer port, Übertragungsgeschwindigkeit bis zu 1MB/s
- 40-Pin-DIL-Sockel für Bauteile von 8 bis 40 Pins mit der Packungsbreite von 300/600 mil
- 2 D/A Generierungskonverter für das Generieren von VCCP, VPP, gesteuerte Flanken- und Abfallflankensteilheit
- VCCP-Umfang: 0V bis 6,5V/150mA
- VPP-Umfang: 0V bis 15V/100mA
- Selbstkalibrierung und Selbsttest
- Spezialisierter Analogpintreiber (GND/VCC/VPP) für MCS51 und AVR
- Eine "Reserve"-GND-, VCC- und VPP-Anlage, die künftig nach Bedarf des pinmäßig nicht kompatiblen Bauteils verwendbar ist
- TTL-Pintreiber unterstützt H, L, CLK, pull-up, pull-down und Lesen auf allen Pins
- Steckverbinder für In-circuit-Serial-Programming (ISP)
- Selbsttest

Unterstützte Bauteile:

- Mikroprozessoren der MCS51-Reihe: 87Cxxx, 89Cxxx, 89Sxxx von Atmel, Atmel W&M, Intel, Philips, ..., 8/40 Pins
- Mikroprozessoren Atmel AVR: ATtiny, AT90Sxxxx, Atmega, 8 bis 40 Pins, Parallel- und Serienmodus
- Serien-E(E)PROM: 24Cxxx-, 24Fxxx-, 25Cxxx-, 59Cxxx-, 85xxx-, 93Cxxx-Reihen (*)

Anmerkung: *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

Programmiergeschwindigkeit

Anmerkung: *Die Programmierzeiten sind von der PC-Geschwindigkeit, vom Typ des LPT-Ports, von verfügbaren Quellen des Steuersystems sowie von Störfaktoren in der Umgebung abhängig. Zum Vergleich werden die Zeiten für zwei PC-Konfigurationen angeführt:*
Diese Zeiten sind nur illustrativ. Aktuelle Zeiten finden Sie auf www.elnec.com (das Obermenü PRODUCTS, das Linksmenü Typ des Programmers).

Bauteil	Operation	Zeit A	Zeit B
AT89C52	Programmieren und Vergleich	17,5 sek	15,5 sek
T87C5111	Programmieren und 2 x Vergleich	45 sek	25 sek

Messungsvoraussetzungen:
Zeit A: *Pentium MMX, 250 MHz, ECP/EPP, WIN98.*

Zeit B: *Athlon, 750 MHz, ECP/EPP, auf PCI-Bit-Bus, WIN98*

Operationen

- **Standardoperationen:**
  - Intelligente Auswahl des Bauteils nach Typ, Hersteller oder Namenfragments des Bauteils
  - Leertest
  - Lesen
  - Programmieren
  - Vergleich
  - Vergleich und Speicherung der Unterschiede auf die Festplatte
  - Löschen
  - Programmieren der Konfigurations- und Sicherheisbits
- **Sicherheitsoperationen:**
  - Einsetzungstest
  - Kontakttest ZIF-Bauteil
  - Identifizierungs-(ID)-Byte-Test
- **Sonderoperationen:**
  - Statistik
  - Count-Down-Modus
  - automatische Zunahme der Seriennummer

Puffer-Operationen:

- view/edit, find/replace
- fill, copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, Tektronix, ASCII-SPACE-HEX

Die auf den PC gestellten Anforderungen:

*Siehe **Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 12...15V DC, max. 500mA
- Energieaufwand: max. 5W
- Abmessungen: 132x66x30 mm
- Gewicht: ca. 200 g
- Temperaturbereich: 5°..40°C
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- T51prog-Programmer
- Verbindungskabel für die Verbindung des PCs mit dem Programmer

- ISP-Kabel
- Netzstromquelle 220VAC/12VDC, 500 mA, nicht stabilisiert
- Diagnostikkopf für den Selbsttest des Programmers
- Staubschutzdeckel für ZIF-Sockel
- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren
- neue Software-Versionen kostenlos

51&AVRprog

Einleitung

Der 51&AVRprog-Programmer ist ein kleiner und leistungsfähiger **Programmer** für Monochip-Mikroprozessoren der MCS51- und Atmel AVR-Reihen. Er programmiert auch Serien-EEPROMs mit IIC-, Microwire- und SPI-Schnittstellen. Der Programmer ist für all jene bestimmt, die mit den Prozessoren der MCS51-Gruppe, bzw. mit den Atmel- AVR-Prozessoren arbeiten und für die ein Universalprogrammer entweder zu teuer ist oder wegen seinen Abmessungen für den jeweiligen Verwendungszweck nicht eingesetzt werden kann. Der Programmer ist mit einem 40 PIN-ZIF-Sockel ausgestattet.

Der 51&AVRprog-Programmer ist zum Anschluss an einen IBM-kompatiblen PC vorgesehen. Er kann jedoch auch an einen PC AT und höhere Typen, von PC Desktop bis zum Palmtop, angeschlossen werden. Der Programmer wird zum Computer über einen **parallelen Standardport** angeschlossen, der normalerweise zum Anschluss eines Druckers verwendet wird. Beim Anschließen des 51&AVRprog-Programmers an einen PC ist es also nicht notwendig, den PC auszuschalten oder auseinanderzunehmen.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, nach Herstellungsbetrieben oder nach den vorgeschriebenen Masken gewählt werden, gewährleistet eine sehr einfache und bequeme Orientierung in den unterstützten Bauteilen. Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen) sind um Testfunktionen (illegal bit test, Überprüfung der Lage auf die Richtigkeit) sowie um die Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt.

Das Steuerprogramm gewährleistet komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Datei-formates, und nimmt die erforderliche Datenkonversion vor.

Für den 51&AVRprog-Programmer stehen **Sockelkonverter** vom DIL auf PLCC und SOIC zur Verfügung.

Hinweis: *Achten Sie bitte darauf, dass die Verbindung zwischen den PINs JP2, JP3 und JP4 (diese sind nach der Abnahme der Bodenplatte des Programmers zugänglich) richtig eingestellt wird. Bei den AVR-Bauteilen muss die Verbindung auf den JP3/JP4-PINs, bei den MCS51-Bauteilen auf den JP2/JP3-PINs liegen. Die Default-Stellung der Verbindung befindet sich auf den JP2/JP3-PINs (für die MCS51-Bauteile).*

Wie wird der 51&AVRProg-Programmer am PC angeschlossen?

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem 51&AVRprog-Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel für den Drucker heraus und an seiner Stelle schließen Sie das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an den freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll. Der Austausch des Kabels zum Drucker für den Kabel zum Programmer ist etwas umständlich, es wird jedoch nicht empfohlen, den 51&AVRprog-Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Druckerumschalter ist nicht möglich.

Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“, der 51&AVRprog -Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren.
Starten Sie jetzt das Steuerprogramm für den 51&AVRprog-Programmer.

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des 51&AVRprog-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

- **Anschluss:** *ZUERST das Kabel zum PC und DANN die Speisung.*
- **Abkopplung:** *ZUERST die Speisung und DANN das Kabel zum PC.*

Probleme der 51&AVRProg ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der 51&AVRprog ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

Handhabung mit den zu programmierenden Bauteilen

Nachdem Sie das Bauteil gewählt haben, legen Sie dieses in den ZIF-Sockel ein (Hebel in der Stellung oben) und legen Sie den

Hebel um, um den Kontakt zwischen dem Bauteil und dem ZIF-Sockel herzustellen.
Die richtige Lage des zu programmierenden Bauteils im ZIF-Sockel ist aus dem Schema neben dem ZIF-Sockel ersichtlich. Das zu programmierende Bauteil soll in den ZIF-Sockel nur in dem Falle eingelegt oder aus dem ZIF-Sockel herausgenommen werden, wenn die LED „BUSY“ nicht mehr leuchtet.

Warnung*: Der 51&AVRprog-Programmer ist mit keinen Sicherheitsschaltungen ausgestattet, die den Inhalt des zu programmierenden Bauteils in kritischen Zuständen schützen (z.B. Stromausfall, Unterbrechung der Kommunikation mit dem PC). Das Bauteil kann ebenfalls in dem Falle zerstört werden, dass während des Programmierens der Ablauf des Steuerprogramms beabsichtigt unterbrochen wird (RESET, Computer ausschalten...) oder wenn das Bauteil aus dem ZIF-Sockel herausgenommen wird. Durch falsches Einlegen des Bauteils in den ZIF-Sockel kann das zu programmierende Bauteil entweder beschädigt oder zerstört werden.*

Technische Spezifikation

Sockel, Pin-Treiber und DAC

- DIL ZIF-Sockel mit 40 Pins (Zero Insertion Force) für Bauteile mit 8, 20, 40 Pins und der Packungsbreite von 300/600 mil
- Spezialisierter Analogpintreiber (GND/VCC/VPP) für MCS51- und AVR-Bauteile
- TTL-Treiber unterstützt H, L sowie Lesen auf allen PINs

Unterstützte Bauteile

- Mikroprozessoren der 51-Reihe: 87Cxxx, 89Cxxx, 89Sxxx von Atmel, Intel, Philips...mit 20/40 Pins
- Mikroprozessoren Atmel AVR: AT90Sxxxx (Parallelmodus), mit 8/20/40 Pins
- Serien-E(E)PROM: 24Cxxx, 24Fxxx, 25Cxxx, 59Cxxx, 85xxx, 93Cxxx (*)

Anmerkungen:

- *Das Programmieren von Serien-EEPROMs ist nicht auf die Geschwindigkeit ausgelegt!*
- *Die Hardware des Programmers ermöglicht nicht, einige der neuesten MCS51/AVR-Bauteile zu programmieren.*
- *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

Operationen

- **Standardoperationen:**
  - Leertest
  - Lesen
  - Programmieren
  - Vergleich
  - Vergleich und Speicherung der Unterschiede auf die Festplatte
  - Löschen
  - Programmieren der Konfigurations- und Sicherheitsbits
- **Sicherheitsoperationen:**
  - Identifizierungs-(ID)- Byte-Test
- **Sonderoperationen:**
  - automatische Zunahme der Seriennummer

Puffer-Operationen:

- view/edit, find/replace
- fill/copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, Tektronix, ASCII-SPACE-HEX

Die auf den PC gestellten Anforderungen:

*Siehe **Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 15V...20V DC/200mA
- Energieaufwand: max. 3W
- Abmessungen: 132x66x30 mm
- Gewicht: ca. 200g
- Temperaturbereich: 5°..40°C
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- 51&AVRProg-Programmer
- Verbindungskabel für die Verbindung des PCs mit dem Programmer
- Netzstromquelle: 220VAC/15VDC, 500mA nicht stabilisiert
- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren
- neue Software-Versionen kostenlos

PIKprog+

Einleitung

PIKprog+ stellt die neue Generation von spezialisierten ELNEC-Programmern, die durch Windows 95/98/Me/NT/2000/XP gesteuert werden. Der Programmer wurde so entwickelt, um alle Anforderungen an professionelle Programmer von seiten der Firma Microchip™ zu erfüllen.

Der PIKprog+-Programmer ist ein kleiner, leistungsstarker und extrem schneller Serviceprogrammer für alle Monochip-Mikroprozessoren der Microchip™ PICmicro®-Reihe (bis zu 40 Pins DIL), der zum stationären und mobilen Gebrauch bestimmt ist. Der Programmer programmiert auch Serien-EEPROMs mit IIC-, Microwire- und SPI-Schnittstellen. Der Programmer ist auch mit einem In-circuit-Steckverbinder zum serienmäßigen Programmieren (ISP) ausgestattet, was seinen Gebrauchswert noch steigert. Der PIKprog+-Programmer ist für all jene bestimmt, die mit Mikroprozessoren Microchip™ PICmicro®-Reihe arbeiten, und für die ein Universalprogrammer ungeeignet ist, sei es aus Preisgründen oder aus Gründen der Mobilität und Abmessungen.

Der Programmer ist ein technisch hervorragend ausgearbeitetes Produkt, das zuverlässiges und schnelles Programmieren zu einem sehr wettbewerbsfähigen Preis bietet. In seiner Klasse weist er sicher eins der besten Preis-Leistungs-Verhältnisse vor.

Der Programmer programmiert die Bauteile aus dem Grund sehr schnell, weil die leistungsstarke Hardware auf FPGA-Basis und ECP/EPP(IEEE1284)-Unterstützung des Parallelports verwendet wurde. Der Programmer ist in seiner Klasse viel schneller, als die meisten Programmer der Mitbewerber.

Der **PIKprog+**-Programmer ist für den Anschluss an alle IBM-kompatible PCs vorgesehen. Er kann zum PC AT und höheren Typen, vom PC Desktop bis zum Palmtop über einen parallelen Standardport angeschlossen werden, der normalerweise zum Anschluss eines Druckers verwendet wird. Beim Anschließen des Programmers muss der PC natürlich weder ausgeschaltet noch auseinandergenommen werden.

Der **PIKprog+** hat 40 leistungsstarke TTL-Pin-Treiber, die H, L, pull-up, pull-down und die Lesemöglichkeit für jedes Pin des Sockels bieten. Verwendete fortgeschrittene hochwertige und Hochgeschwindigkeitsbauteile bieten ausgezeichnete Signale ohne Überschwingungen zum Programmieren aller unterstützten Bauteile. Der Pin-Treiber bietet eine ausreichend niedrige Speisespannung, um das Programmieren von allen gegenwärtigen (1,8 V) sowie künftigen Low-voltage-Bauteilen zu ermöglichen.

Das zu programmierende Bauteil ist vor einer Beschädigung durch Fehler der Bedienung durch leistungsstarke Sicherheitsmaßnahmen geschützt: Kontrolle des **richtigen Einsetzens der Bauteils** in den Sockel, Kontrolle **des richtigen**

Kontaktes der Leitungen, und natürlich die Kontrolle **der elektronischen Identifizierung des Bauteils.**

Der Vergleich des Programmierens erfolgt durch den Programmer selbst bei **Stromversorgungsgrenzwerten**, was natürlich die Richtigkeit des Programmierablaufes erhöht und die Aufbewahrung der in dem Bauteil eingespeicherten Daten über längere Zeit gewährleistet.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, Herstellungsbetrieben oder den vorgeschriebenen Masken gewählt werden, gewährleistet eine sehr einfache und bequeme Orientierung in den unterstützten Bauteilen. Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen des Bauteils, Löschen) sind um Testfunktionen (illegal bit test, Überprüfung der Lage auf die Richtigkeit) sowie um die Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt.

Das Steuerprogramm gewährleistet die komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Dateiformats, und nimmt die erforderliche Datenkonversion vor.

Die **Zunahmefunktion** gewährleistet, dass jedes Mal, bevor ein neues Bauteil eingesetzt wird, im Puffer die Erhöhung der Seriennummer erfolgt. Die automatische Zunahmefunktion ermöglicht, dass die Seriennummer, bzw. die individuelle Identifikation des zu programmierenden Bauteils ebenfalls aus der jeweiligen Datei gelesen werden kann.

Zum Programmieren von Bauteilen in anderen als in DIL-Packungen steht ein breites Sortiment an Konvertern von DIL auf PLCC, SOIC und anderen Packungen zur Verfügung.

Wie wird der PIKprog+ -Programmer am PC angeschlossen?

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem PIKprog+-Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel für den Drucker heraus und schließen Sie an seiner Stelle das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an den freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll. Der Austausch des Kabels zum Drucker für den Kabel zum Programmer ist etwas umständlich, es wird jedoch nicht empfohlen, den PIKprog+-Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Druckerumschalter ist nicht möglich.

Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“ auf und der PIKprog+-Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren.
Jetzt starten Sie das Steuerprogramm für den PIKprog+-Programmer.

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des PIKprog+-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

- **Anschluss:** *ZUERST das Kabel zum PC und DANN die Speisung.*
- **Abkopplung:** *ZUERST die Speisung und DANN das Kabel zum PC.*

Probleme der PIKprog+ ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der PIKprog+ ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

In-system Serial Programming mit dem PIKprog+-Programmer

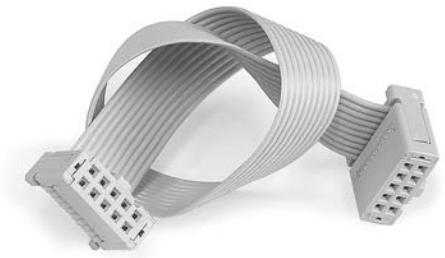
Definitionen, Empfehlungen und Hinweise sind im Teil **Gemeinsame Hinweise/ISP** angeführt.

Beschreibung des ISP-Steckverbinders des PIKprog+-Programmers

Pin	**Bedeutung**
2, 3, 4, 6, 8	H/L/Lesen
1	VCCP für das Zielbauteil
7, 9	GND
10	H/L/Lesen, VPP auf MCLR\
5	NC

Die konkrete Bedeutung der Pins des ISP-Steckverbinders hängt vom zu programmierenden Bauteil ab und ist im Fenster **Informationen über das Bauteil** dargestellt. Die Bedeutung der PINs entspricht den Empfehlungen aus der Applikationshinweise der Firma Microchip In-Curcuit Serial Programming™ (ICSP™) Guide. Diese Applikationshinweise können Sie auf www.elnec.com im Teil Applikationshinweise finden (Das Obermenü SUPPORT, das Linksmenü Application notes).

Anmerkung: *Der Pin 1 ist am ISP-Steckverbinder des Kabels durch einen dreieckigen Einschnitt gekennzeichnet*.



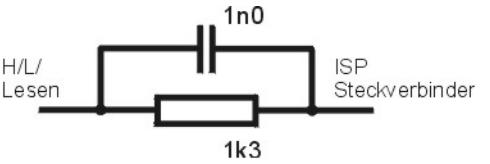
ISP-Kabel des PIKprog+-Programmers

Hinweis:

- ***Wenn*** *Sie den PIKprog+ als* ***ISP-Programmer*** *verwenden,* ***legen Sie*** *das Bauteil* ***nicht in den ZIF-Sockel.***
- ***Wenn*** *Sie Bauteile im* ***Sockel*** *programmieren,* ***stecken Sie das ISP-Kabel nicht in den ISP-Steckverbinder.***
- *Verwenden Sie ausschließlich das* ***beigelegte ISP-Kabel.*** *Verwendung eines anderen ISP-Kabels (anderes Material, andere Länge, usw.) kann Grund für ein unzuverlässiges Funktionieren sein.*

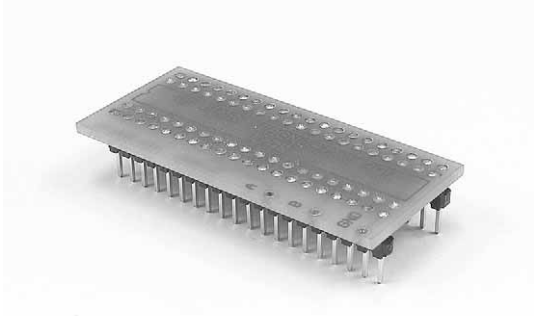
- ***PIKprog+*** *kann nur ein programmierbares Bauteil speisen, aber die Zielanlage* ***kann nicht den PIKprog+ speisen.***
- ***PIKprog+*** *führt die Programmierspannung an das Zielbauteil und prüft ihren Wert (das Zielsystem kann die Programmierspannung beeinflussen). Wenn die Programmierspannung von der gewünschten abweicht, wird die Arbeit mit dem Bauteil unterbrochen.*

Anmerkung: *Treiber des PIKprog+-Programmers zum H/L/Lesen*



Selbsttest und Programmer-Eichung

Haben Sie das Gefühl, dass Ihr Programmer nicht ganz nach Ihren Erwartungen arbeitet, führen Sie bitte den Selbsttest des Programmers mit Hilfe des Diagnostikkopfes (Diagnostic POD) durch. Der Diagnostikkopf wird serienmäßig zusammen mit dem PIKprog+-Programmer geliefert. Um während der Arbeit mit dem PIKprog+-Programmer die optimalen Ergebnisse auf dem selben Niveau zu halten, wird es empfohlen, einmal in **sechs** Monaten (falls der Programmer sehr oft eingesetzt wird, einmal in drei Monaten) den Programmer zu testen und die Eichung zu überprüfen. Folgen Sie dabei den Anweisungen des Steuerprogramms im **Diagnostik-Menü**.



Handhabung mit den zu programmierenden Bauteilen

Nachdem Sie das Bauteil gewählt haben, legen Sie dieses in den ZIF-Sockel ein (Hebel in der Stellung oben) und legen Sie den Hebel um, um den Kontakt zwischen dem Bauteil und dem ZIF-Sockel herzustellen.
Die richtige Lage des zu programmierenden Bauteils im ZIF-Sockel ist aus dem Schema neben dem ZIF-Sockel ersichtlich. Das zu programmierende Bauteil soll in den ZIF-Sockel nur in dem Falle eingelegt oder aus dem ZIF-Sockel herausgenommen werden, wenn die LED „BUSY“ nicht mehr leuchtet.

Warnung*: Der PIKprog+-Programmer ist mit keinen Sicherheitsschaltungen ausgestattet, die den Inhalt des zu programmierenden Bauteils in kritischen Zuständen schützen würden (z.B. Stromausfall, Unterbrechung der Kommunikation mit dem PC). Das Bauteil kann ebenfalls in dem Falle zerstört werden, wenn während des Programmierens der Ablauf des Steuerprogramms plötzlich unterbrochen wird (RESET, Computer ausschalten...) oder wenn das Bauteil aus dem ZIF-Sockel herausgenommen wird. Durch falsches Einlegen des Bauteils in den ZIF-Sockel kann das zu programmierende Bauteil entweder beschädigt oder zerstört werden.*

Technische Spezifikation

HARDWARE

Programmer

- 2 D/A Generierungskonverter für das Generieren von VCCP, VPP, gesteuerte Flanken- und Abfallflankensteilheit
- VCCP-Umfang: 0V bis 7V/250mA
- VPP-Umfang: 0V bis 15V/100mA
- Mit Hilfe von FPGA realisierter IEEE 1284 slave printer port, Übertragungsgeschwindigkeit bis zu 1MB/s
- Selbstkalibrierung und Selbsttest

Sockel und Pin-Treiber

- 40-Pin-DIL-Sockel für Bauteile von 8 bis 40 Pins mit der Packungsbreite von 300/600 mil
- Pin-Treiber: 40 TTL-Pin-Treiber, spezialisierter Analogpintreiber (GND/VCC/VPP) für PICMicro®-Bauteile
- eine "Reserve"-GND-, VCC- und VPP-Anlage, die künftig nach Bedarf des pinmäßig nicht kompatiblen Bauteils verwendbar ist
- der mit Hilfe von FPGA realisierter TTL-Pin-Treiber unterstützt H, L, CLK, pull-up, pull-down und Lesen auf allen Pin-Treiber-Pins, das H-Niveau ist von 1,8V bis 5V einstellbar
- Steckverbinder für In-circuit-Serial-Programming (ISP)
- Test der Leitfähigkeit (ZIF des Programmers – Bauteil) für jeden Pintreiber-Pin

Unterstützte Bauteile:

- Mikroprozessoren der Microchip™ PICmicro®-Reihe: 12xxx-, 14xxx-, 16xxx-, 17xxx-, 18xxx-Reihen, 8 bis 40 Pins (*1), Parallel- und Serienmodus. Der PIKprog+-Programmer erfüllt alle Empfehlungen der Firma Microchip™ bezüglich professioneller Programmer.
- Serien-EEPROM: 24Cxxx-, 24Fxxx-, 25Cxxx-, 59Cxxx-, 85xxx-, 93Cxxx-Reihen

Anmerkungen:

- *(*1) – für non-DIL-Packungen und Bauteile mit mehr als 40 Pins stehen entsprechende Packungskonverter zur Verfügung.*
- *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

Programmiergeschwindigkeit

Anmerkung: *Die Programmierzeiten sind von der PC-Geschwindigkeit, vom Typ des LPT-Ports, von verfügbaren Quellen des Steuersystems sowie von Störfaktoren in der Umgebung*

abhängig. Zum Vergleich werden die Zeiten für zwei PC-Konfigurationen angeführt:
Diese Zeiten sind nur illustrativ. Aktuelle Zeiten finden Sie auf www.elnec.com (das Obermenü PRODUCTS, das Linksmenü Typ des Programmers).

Bauteil	Operation	Zeit A	Zeit B
PIC16C67	Programmieren und Vergleich	13 sek	11 sek
PIC18F452	Programmieren und Vergleich	11 sek	9 sek

Messungsvoraussetzungen:
Zeit A: *Pentium MMX, 250 MHz, ECP/EPP, WIN98.*
Zeit B: *Athlon, 750 MHz, ECP/EPP, auf PCI-Bit-Bus, WIN98*

Operationen

- **Standardoperationen:**
  - Intelligente Auswahl des Bauteils nach Typ, Hersteller oder Namenfragments des Bauteils
  - Leertest, Lesen, Vergleich
  - Programmieren
  - Vergleich und Speicherung der Unterschiede auf die Festplatte
  - Löschen
  - Programmieren der Konfigurations- und Sicherheisbits
  - illegal bit test
  - Prüfsumme
- **Sicherheitsoperationen:**
  - Einsetzungstest
  - Kontakttest ZIF-Bauteil
  - Identifizierungs-(ID)-Byte-Test
- **Sonderoperationen:**
  - automatische Zunahme der Seriennummer
  - Statistik
  - Count-Down-Modus

Puffer-Operationen:

- view/edit, find/replace
- fill, copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Datei laden/speichern

- keine Ladezeit, weil der Programmer mit dem PC gesteuert ist
- automatische Identifikation des Dateiformats

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, Tektronix, ASCII-SPACE-HEX

Die auf den PC gestellten Anforderungen:

Siehe ***Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 12...15V DC, max. 0,5 A
- Energieaufwand: max. 5W, ca. 0,5W sleep
- Abmessungen: 132x66x30 mm
- Gewicht: ca. 200 g (ohne Speisequelle)
- Temperaturbereich: 5°..40°C
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- PIKprog+-Programmer
- Verbindungskabel für die Verbindung des PCs mit dem Programmer
- ISP-Kabel
- Netzstromquelle 220VAC/12VDC, 500 mA, nicht stabilisiert
- Diagnostikkopf für den Selbsttest des Programmers
- Staubschutzdeckel für ZIF-Sockel
- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren
- neue Software-Versionen kostenlos

PIKprog

Einleitung

PIKprog ist ein kleiner und leistungsfähiger Programmer für das Programmieren der Microchip/PICmicro-Singlechip-Mikroprozessoren. Ebenfalls kann er zum Programmieren der serienmäßigen EEPROMs mit der IIC-, Microwire-, aber auch SPI-Schnittstellen eingesetzt werden. Der Programmer ist mit einem 40-Pin DIL ZIF-Sockel versehen.

Der PIKprog-Programmer ist für den Anschluss an einen IBM-kompatiblen PC vorgesehen. Er kann an PC AT und höhere Typen angeschlossen werden, vom PC Desktop bis zum Palmtop. Der Programmer wird am Computer über einen **parallelen Standardport** angeschlossen, der normalerweise zum Anschluss eines Druckers verwendet wird. Beim Anschließen des PIKprog-Programmers an einen PC ist es also nicht notwendig, den PC auszuschalten oder auseinanderzunehmen.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Die Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, nach Herstellungsbetrieben oder nach der vorgeschriebenen Masken gewählt werden, gewährleistet eine sehr einfache und bequeme Orientierung in den unterstützten Bauteilen. Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen des Bauteils) sind um Testfunktionen (illegal bit test, Überprüfung der Lage auf die Richtigkeit) sowie um die Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt.

Das Steuerprogramm gewährleistet komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Dateiformates, und nimmt die erforderliche Datenkonversion vor.

Für den PIKprog-Programmer stehen **Sockelkonverter** vom DIL auf SOIC zur Verfügung.

Wie wird der PIKprog-Programmer am PC angeschlossen?

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem PIKprog-Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel für den Drucker heraus und an seiner Stelle schließen Sie das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an den freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll.
Der Austausch des Kabels zum Drucker für den Kabel zum Programmer ist etwas umständlich, es wird jedoch nicht empfohlen, den PIKprog-Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Druckerumschalter ist nicht möglich.

Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“, der PIKprog-Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren.
Jetzt starten Sie das Steuerprogramm für den PIKprog-Programmer.

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des PIKprog-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

- ***Anschluss:** ZUERST das Kabel zum PC und DANN die Speisung.*
- ***Abkopplung:** ZUERST die Speisung und DANN das Kabel zum PC.*

Probleme der PIKprog ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der PIKprog ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

Handhabung mit den zu programmierenden Bauteilen

Nachdem Sie das Bauteil gewählt haben, legen Sie dieses in den ZIF-Sockel ein (Hebel in der Stellung oben) und legen Sie den

Hebel um, um den Kontakt zwischen dem Bauteil und dem ZIF-Sockel herzustellen.
Die richtige Lage des zu programmierenden Bauteils im ZIF-Sockel ist aus dem Schema neben dem ZIF-Sockel ersichtlich. Das zu programmierende Bauteil darf in den ZIF-Sockel nur in dem Falle eingelegt oder aus dem ZIF-Sockel herausgenommen werden, falls die LED „BUSY“ nicht mehr leuchtet.

Warnung: *Der PIKprog-Programmer ist mit keinen Sicherheitsschaltungen ausgestattet, mit denen der Inhalt des zu programmierenden Bauteils in kritischen Zuständen geschützt wird (z.B. Stromausfall, Unterbrechung der Kommunikation mit dem PC). Das Bauteil kann ebenfalls in dem Falle zerstört werden, dass während des Programmierens der Ablauf des Steuerprogramms beabsichtigt unterbrochen wird (RESET, Computer ausschalten...) oder wenn das Bauteil aus dem ZIF-Sockel herausgenommen wird. Durch falsches Einlegen des Bauteils in den ZIF-Sockel kann das zu programmierende Bauteil entweder beschädigt oder zerstört werden.*

Technische Spezifikation

Sockel, Pin-Treiber und DAC

- DIL ZIF-Sockel mit 40 Pins (Zero Insertion Force) für die Bauteile bis 40 Pins und mit Packungsbreite von 300/600 mil
- TTL-Treiber unterstützt H, L sowie Lesen auf allen PINs

Unterstützte Bauteile

- Mikroprozessoren Microchip PICmicro: 12xxx, 14xxx, 16xxx, 18xxx, mit 8 bis 40 Pins
- Serien-EEPROM: 24Cxxx-, 24Fxxx-, 25Cxxx-, 59Cxxx-, 85xxx-, 93Cxxx(*)-Reihen

Anmerkungen:

- *Das Programmieren der Serien-EEPROMs ist nicht in bezug auf Geschwindigkeit optimalisiert!*
- *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

Der PIKprog-Programmer erfüllt die Empfehlungen des Herstellers für professionelle Programmer.

Operationen

- **Standardoperationen:**
  - Leertest
  - Lesen
  - Programmieren
  - Vergleich
  - Vergleich und Einsetzen der Unterschiede auf die Festplatte
  - Löschen
  - Programmieren der Konfigurations- und Sicherheitsbits
- **Sonderoperationen:**
  - automatische Zunahme der Seriennummer

Puffer-Operationen:

- view/edit, find/replace
- fill/copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, Tektronix, ASCII-SPACE-HEX

Die auf den PC gestellten Anforderungen:

*Siehe **Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 15-20V DC/max. 200mA
- Energieaufwand: max. 3W
- Abmessungen: 132x66x30 mm
- Gewicht: ca. 200g
- Temperaturbereich: 5°..40°C
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- PIKprog-Programmer
- Verbindungskabel für die Verbindung des PCs mit dem Programmer
- Netzstromquelle: 220VAC/15VDC, 500mA nicht stabilisiert
- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren
- neue Software-Versionen kostenlos

SEEprog

Einleitung

SEEprog ist ein universeller Programmer für alle serienmäßigen EEPROMs in einer 8-Pin-DIL-Packung mit IIC-, SPI-, Microwire-Schnittstelle. Mit seiner Hilfe können auch z. B. Digitalthermometer und LV-Bauteile (3,3V) programmiert werden.

SEEprog-Programmer ist zum Anschluss an IBM-kompatible PCs vorgesehen. Er kann an PC AT und höhere Typen angeschlossen werden, vom PC Desktop bis zum Palmtop. Der Programmer wird am Computer über einen **parallelen Standardport** angeschlossen, der normalerweise zum Anschluss eines Druckers verwendet wird. Beim Anschließen des Seeprog-Programmers an einen PC ist es also nicht notwendig, den PC auszuschalten oder auseinanderzunehmen.

Benutzerfreundliche Software ermöglicht den Programmer einfach zu bedienen. Auswahl der entsprechenden Bauteile, die nach einzelnen Klassen, nach Herstellungsbetrieben oder nach den vorgeschriebenen Masken gewählt werden, gewährleistet eine sehr einfache und bequeme Orientierung in den unterstützten Bauteilen. Standardfunktionen für die Handhabung des Bauteils (Lesen, Leertest, Programmieren, Vergleich, Löschen eines Bauteils) sind um Sonderfunktionen (automatischer Programmierstart nach dem Einlegen des Bauteils in den Sockel) ergänzt.

Das Steuerprogramm gewährleistet komfortable Bedienung der Dateien, inkl. der automatischen Erkennung des Dateiformates, und nimmt die notwendige Datenkonversion vor.

Für den SEEprog-Programmer stehen **Sockelkonverter** vom DIL auf SOIC zur Verfügung.

Die Programmer-Parameter – seine Zuverlässigkeit und Schnelligkeit – entsprechen hohen Anforderungen auch bei einem eventuellen Einsatz in der Produktion.

Wie wird der SEEprog-Programmer am PC angeschlossen?

Schalten Sie Ihren PC aus und schließen Sie das Verbindungskabel PC<->Programmer, das zusammen mit dem SEEprog-Programmer geliefert wurde, an den Computer an. Benutzen Sie dabei den Steckverbinder, der für den Anschluss des Druckers vorgesehen ist (Printerport). Sollte an Ihrem Computer lediglich ein Printerport vorhanden sein, ziehen Sie aus dem Computer das Kabel für den Drucker heraus und an seiner Stelle schließen Sie das Kabel für den Programmer an. Sollte Ihr Computer über mehrere Printerports verfügen, schließen Sie den Programmer immer an den freien Printerport an. Das andere Kabelende schließen Sie an den Programmer an. Beide Steckverbinder werden in Gegenstücke aufgesteckt und mit diesen zusammengeschraubt. Hinweis: Passen Sie bitte vor allem auf den Steckverbinder auf, der in den Programmer eingeschoben werden soll.
Der Austausch des Kabels zum Drucker für den Kabel zum Programmer ist etwas umständlich, es wird nicht empfohlen, den SEEprog-Programmer über einen mechanischen Druckerumschalter zu betreiben. Der Betrieb über einen elektronischen Druckerumschalter ist nicht möglich.

Schließen Sie die Stromquelle an die Netzsteckdose und den Stecker an den Programmer an. Jetzt schalten Sie Ihren PC ein. Auf dem Programmer leuchtet die LED „POWER“, der SEEprog-Programmer ist bereit, mit dem Steuerprogramm des PC zu kommunizieren.
Starten Sie jetzt das Steuerprogramm für den SEEprog-Programmer.

Hinweis! *Sollten Sie sich entscheiden, beim Anschließen oder Abkopplung des Seeprog-Programmers den PC nicht auszuschalten, halten Sie bitte folgendes Vorgehen ein:*

- **Anschluss:** *ZUERST das Kabel zum PC und DANN die Speisung.*
- **Abkopplung:** *ZUERST die Speisung und DANN das Kabel zum PC.*

Probleme der SEEprog ⇔ PC-Verbindung, Fehlerbehebung

Falls Sie Probleme mit der SEEprog ⇔ PC-Verbindung haben, lesen Sie bitte ***Gemeinsame Hinweise***.

Handhabung mit den zu programmierenden Bauteilen

Nachdem Sie das Bauteil gewählt haben, legen Sie dieses in den ZIF-Sockel ein (Hebel in der Stellung oben) und legen Sie den

Hebel um, um den Kontakt zwischen dem Bauteil und dem ZIF-Sockel herzustellen.
Die richtige Lage des zu programmierenden Bauteils im ZIF-Sockel ist aus dem Schema neben dem ZIF-Sockel ersichtlich. Das zu programmierende Bauteil darf in den ZIF-Sockel nur in dem Falle eingelegt oder aus dem ZIF-Sockel herausgenommen werden, falls die LED „BUSY“ nicht mehr leuchtet.

Warnung: *Der SEEprog-Programmer ist mit keinen Sicherheitsschaltungen ausgestattet, mit denen der Inhalt des zu programmierenden Bauteils in kritischen Zuständen geschützt wird (z.B. Stromausfall, Unterbrechung der Kommunikation mit dem PC). Der Bauteil kann ebenfalls in dem Falle zerstört werden, dass während des Programmierens der Ablauf des Steuerprogramms beabsichtigt unterbrochen wird (RESET, Computer ausschalten...) oder wenn das Bauteil aus dem ZIF-Sockel herausgenommen wird. Durch falsches Einlegen des Bauteils in den ZIF-Sockel kann das zu programmierende Bauteil entweder beschädigt oder zerstört werden.*

Technische Spezifikation

Sockel, Pin-Treiber

- 24-Pin-DIL ZIF Sockel für Bauteile mit Packungsbreite von 300 mil (*1)
- Jeder Pin des Sockels kann in Zustände low und pull-down eingestellt werden.
- Aus jedem Pin des Sockels kann gelesen werden.
- Unterstützung der Low-Voltage-Bauteile (Speisung von 3,3V)

Anmerkung:

*(*1) Für die Arbeit mit Serien-EEPROMs sind 8 Pins, die dem Sockelhebel am nächsten sind, verwendet, ungenutzte Pins sind nicht angeschlossen.*

Unterstützte Bauteile

Programmer

- EEPROM 12C (24Cxxx)
- EEPROM Microwire (93Cxxx)
- EEPROM SPI (25Cxxx)
- spezielle Bauteile (Digitalthermometer,...)

Anmerkung: *Alle unterstützte Bauteile finden Sie auf www.elnec.com (das Obermenü DOWNLOAD, die Tafel Software for programmers, die Kolonne device list).*

Operationen

- **Standardoperationen:**
  - Leertest
  - Lesen
  - Programmieren
  - Vergleich
  - Löschen
  - Prüfsumme

Puffer-Operationen:

- view/edit, find/replace
- fill/copy, move, byte swap, word/dword split
- Prüfsumme (byte, word)
- print

Zulässige Dateiformate:

- binarisches Originalformat
- HEX: Intel, Intel EXT, Motorola S, MOS, Exormax, ASCII-SPACE-HEX

Die auf den PC gestellten Anforderungen:

*Siehe **Einleitung/Die an den Steuerrechner gestellten Anforderungen***

Allgemeine Anforderungen:

- Stromversorgung: 8-20V DC/ max. 50mA
- Abmessungen: 132x66x30 mm
- Gewicht: ca. 150g
- Temperaturbereich: 5°..40°C
- Feuchtigkeit: 20%..80%, nicht kondensierend

Lieferung enthält:

- SEEprog-Programmer. Sockel zum Programmieren der ZIF-Art (mit Null-Einsetzkraft, 10.000 Einsätze)
- Verbindungskabel für die Verbindung des PCs mit dem Programmer
- Netzstromquelle: 220VAC/12VDC
- Bedienungsanleitung
- Steuerprogramm
- DEVICE PROBLEM REPORT-Formular
- Registrierkarte
- Verpackung

Zusätzliche Dienstleistungen:

- Keep Current
- AlgOR
- kostenlose Beratung (Hotline) im Bereich Programmieren
- neue Software-Versionen kostenlos

Software

Programmer-Software

Zum Lieferumfang des Programmers gehört eine CD mit dem Steuerprogramm, den HelpUtilities und Zusatzinformationen. Der CD-Inhalt kann als DEMO-Version des Steuerprogramms zu den Programmern der Firma ELNEC frei verwendet werden. Abweichungen und Ergänzungen zu der Installationsvorschrift finden Sie unter www.elnec.com.

Software-Installation

Die Software-Installation ist bei unseren Programmern sehr einfach – starten Sie das Installationsprogramm auf der CD. Das Programm begleitet Sie bei der Installation und führt alle Schritte durch, die vor der Inbetriebnahme des Steuerprogramms notwendig sind.
Falls Sie das **DOS**-Betriebssystem verwenden, starten Sie das **Install.exe** Programm.
Falls Sie das **MS Windows-**Betriebssystem benutzen, starten Sie das **Setup.exe** Programm.

Die Programme PG4U.EXE (für DOS) und PG4UW.EXE (für Windows) stellen die Steuerprogramme für alle Programmer der Firma ELNEC dar. Von unserer Firma wird problemloser Lauf unter jedem der o.g. Betriebssysteme gewährleistet. Ebenfalls problemlos erfolgt die Arbeit unter den Betriebssystemen WINDOWS.

Für die DOS-Version des PG4U-Steuerprogramms unter den Betriebssystemen Windows 3.xx/95/98 gelten folgende Empfehlungen:

- Starten Sie das Programm im Text-Modus als **DOS-Applikation** auf dem gesamten Bildschirm (Full Screen). Im graphischen Modus läuft das Programm ziemlich langsam.
- Falls der Programmer im **Hintergrund** arbeiten soll, ist bei WIN95 einzustellen: Programm/Eigenschaften/Sonstiges/ Hintergrund – jedes Mal unterdrücken = nein, WIN3xx: in der entsprechenden PIF-Datei soll die Wahl „Execution“ auf „Background“ eingestellt werden.

Hinweis: *Benutzerbetreuung der DOS-Software für unsere Programmer ist seit 7/2003 untergebrochen worden.*

Neue Versionen des Steuerprogramms

Die neuesten Versionen des Steuerprogramms ermöglichen, den Programmer und seine Eigenschaften in vollem Maß auszunutzen (nähere Informationen – dazu siehe Anlage, Keep-Current-Dienstleistung). Die neueste Version des Steuerprogramms (PG4UARC.EXE/ PG4UWARC.EXE) steht Ihnen kostenlos auf der www-Seite: **www.elnec.com, Teil download** zur Verfügung. Die CD-ROM mit dem Steuerprogramm können wir Ihnen auch per Post gegen eine Bearbeitungsgebühr übersenden.

Installation der neuen Version des Steuerprogramms

Die neue Version (upgrade) des Steuerprogramms steht Ihnen in komprimierter Form als selbstauspackende EXE-Datei (PG4UARC.EXE/PG4UWARC.EXE) zur Verfügung. Die Datei sollte auf das Laufwerk in ein separates Verzeichnis übertragen werden. Nach dem Start des Programms werden dann sämtliche Dateien entpackt. Das weitere Verfahren ist dasselbe, wie das Verfahren bei der Installation des Programms aus der CD-ROM. Nach der Beendigung der Installation des Programms empfehlen wir Ihnen, alle Dateien in dem temporären Verzeichnis zu löschen.

Verwendungsweise des Steuerprogramms

Die Steuerprogramme für unsere Erzeugnisse sind zur Zeit ihrer Auslieferung virenfrei. Aus Sicherheitsgründen ist jedes unserer Steuerprogramme mit einem Kontrollalgorithmus ausgestattet, der das eventl. Vorhandensein eines Computervirus erkennt.

Start des Steuerprogramms

Geben Sie bitte in der MS/PC-DOS-Eingabeaufforderung ein:

PG4U

für Windows: klicken Sie zweimal auf das PG4UW-Icon.

Nach dem Start des Steuerprogramms werden von diesem alle zugänglichen Parallelports automatisch überprüft und der angeschlossene ELNEC-Programmer ausgesucht. Weil das PG4U-Programm von allen Programmern der Firma ELNEC benutzt wird, wird vom Programm das Vorhandensein von allen unterstützten Programmern (JetProg, LabProg+, SmartProg, PREPROM-02aLV, T51prog, 51&AVRprog, PIKprog+, PIKprog und SEEprog) überprüft.

Hinweis: *Nach dem Start des Steuerprogramms wird die Integrität des Steuerprogramms durch Vergleich der erwarteten und der ermittelten Prüfsumme vom Programm selbst überprüft. Ist das Ergebnis der Überprüfung positiv, wird vom Steuerprogramm der Standardbildschirm angezeigt. Das Steuerprogramm wartet auf Ihre Befehle.*

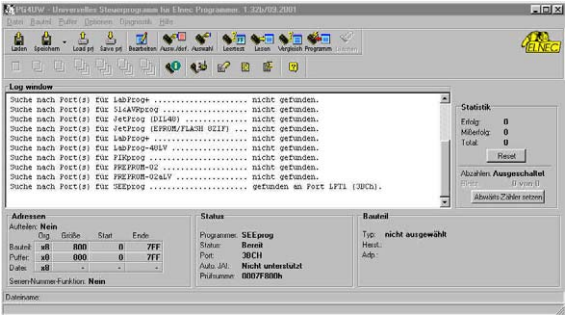
Benutzerbetreuung der DOS-Software für unsere Programmer ist seit 7/2003 untergebrochen worden.

Ist das Steuerprogramm nicht in der Lage, mit dem Programmer zu kommunizieren, erscheint die Fehlermeldung mit möglichen Störungsursachen (der Programmer ist nicht angeschlossen, die Flachstecker sind nicht richtig eingeschoben, Speisungsfehler, Printerport ist nicht kompatibel, oder falsch eingestellt...). Zuerst muss der Fehler behoben und anschließend eine beliebige Taste gedrückt werden. Wurde die Störungsursache nicht behoben, läuft das Programm nach dem Drücken einer beliebiger Taste weiter im

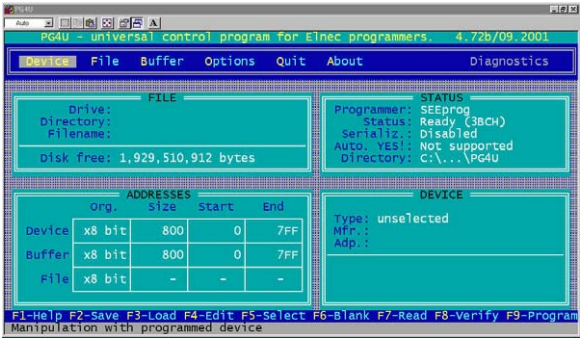
DEMO-Modus, d.h. der Zugriff auf den Programmer ist nicht möglich. Sollten Sie nicht in der Lage sein, die Störungsursache zu beheben, lesen Sie bitte die im Teil **Wenn Probleme auftreten** angeführten Anweisungen. Die Kommunikation mit dem Programmer wird vom Steuerprogramm ebenfalls vor jeder Arbeit mit dem zu programmierenden Bauteil überprüft.

Anwendungs-Bildschirm

Windows-PG4UW-Programm



DOS -PG4U-Programm



Dateikopf	Bezeichnung, Copyright und die Steuerprogramm-Version
Menü	Übersicht- Befehle des Steuerprogramms
Datei-Fenster	Auskünfte über die in den Puffer bereits eingelesene Datei
Status-Fenster	Auskünfte über den Status des Programmers und des Steuerprogramms
Addressen-Fenster	Organisation, Größe, Bauteil-Start-Adresse und Bauteil-End-Adresse, Puffer-Adresse und Datei-Adresse
Bauteil-Fenster	Auskünfte über das bereits ausgewählte Bauteil
Hilfe	kurze Beschreibung des gewählten Befehls

Die einzelnen MENÜ-Punkte werden auf Standardart gewählt – entweder mit dem Cursor und anschließend mit der Bestätigung mit **<Enter>-Taste**, oder mit Hilfe des „fetten" Schriftzeichens des MENÜ-Punktes. Beschreibung der schnellen Steuerung mit den sog. Hotkeys, entnehmen Sie bitte aus der **Hotkeys-Zeile**.

Hinweis: *Daten, die mit Hilfe der Tastatur eingegeben werden, besitzen die HEX-Form. Die Ausnahme ist der ASCII-Block unter der Position „View/Edit Buffer".*

Auflistung der Hotkeys:

<F1>	Hilfe	Hilfe-Inhaltanzeige
<F2>	Speichern	Puffer auf den Laufwerk speichern
<F3>	Laden	Einlesen der Dateien in den Puffer
<F4>	Editieren	Pufferanzeige/-Editieren
<F5>	Auswahl Default	Das Bauteil von 10 letztgewählten Bauteilen anwählen
<Alt+F5>	Auswahl manuell	Das Bauteil nach der Klasse oder nach dem Hersteller anwählen
<F6>	Leertest	Leertest des Bauteils
<F7>	Lesen	Einlesen der Bauteildateien in den Puffer
<F8>	Vergleich	Vegleich der Daten des Bauteils mit dem Puffer
<F9>	Programmieren	Programmierung des gewählten Bauteils
<Alt+Q>	Beenden ohne Speichern	Beendet das PG4U/PG4UW Programm
<Alt+X>	Beenden und Speichern	Beendet das PG4U/PG4UW Programm und speichert die Konfiguration
<Ctrl+F1>	Bauteil-Info	Anzeige der Auskünfte über das gewählte Bauteil
<Ctrl+F2>	Löschen	Füllt den Puffer mit angegebenen Daten
<Ctrl+Shift+F2>	Mit Daten	Füllt den Puffer mit zufälligen Daten

Datei

Dieser Befehl ermöglicht, verschiedene Operationen mit den Dateien, Einstellung und Durchsuchung der Verzeichnisse, Änderung des Pfades und die Modifikation der Start- und End-Adresse des Puffers für das Lesen und Speichern von Dateien im

Binär-, **MOTOROLA-**, **MOS Technology-**, **ASCII space-**, **Tektronix-, JEDEC-, POF-** oder **Intel** HEX-Format durchzuführen.

Datei / Laden

Der Befehl erkennt das Datei-Format und liest die Daten aus der auf der Festplatte gespeicherten Datei in den Puffer ein. Diese Daten werden dann in den Puffer im **Binär-**, **MOTOROLA-**, **Tektronix-**, **MOS Technology-**, **ASCII space-, Intel HEX-, JEDEC-** und **POF-**Formát gespeichert. Das Steuerprogramm speichert die letztgültige Dateimaske. Diese Dateimaske kann in die Konfigurationsdatei mit dem Befehl **Optionen / Optionen Speichern** gespeichert werden.

Mit Hilfe der Check-Box **Automatische Erkennung des Dateiformats** wird die Art der Formatbestimmung der gelesenen Datei definiert. Ist die Check-Box bezeichnet, versucht das Steuerprogramm, das Dateiformat als eines der unterstützten Formaten automatisch zu erkennen. Gelingt es dem Steuerprogramm, das Dateiformat zu erkennen, kommt das Binärformat in Frage. Ist die Check-Box nicht bezeichnet, muss von dem Anwender das gewünschte Dateiformat aus der Liste der unterstützten Dateiformate eingegeben werden. Bei einem Binärformat kann zusätzlich der Wert für den **Puffer**-**Anfang** eingegeben werden, von dem die Pufferanfangsadresse definiert ist. An diese Adresse werden die aus der Datei gelesenen Daten in den Puffer eingegeben.

Datei / Speichern

Mit diesem Befehl werden die im Puffer gebildeten Daten gespeichert. Diese Daten können entweder modifiziert sein, oder sie können aus dem Bauteil gelesen und dann in die Datei auf der Festplatte im gewählten Format gespeichert werden. Außerdem können die Pufferblockanfangs- und Pufferblockendeadressen des zu speichernden Puffers eingegeben werden.
Zur Zeit werden folgende Datei-Formate unterstützt: **Binärformat**, **Intel MOTOROLA-**, **Tektronix-**, **ASCII space-, MOS Technology-, JEDEC-** und **POF-**Format. Die Dateimasken können mit dem Befehl **Optionen / Generelle Optionen** auf der Liste **Möglichkeiten für das Lesen der Datei** eingestellt werden.

Wenn Sie die Schnelltaste **<F2>** drücken, wird dieses Menü angezeigt.

Datei / Projekt laden

Dieser Befehl ermöglicht, ein Projekt einzulesen. Ein Projekt ist eine Spezialdatei, die Angaben über das Bauteil, seine Einstellungen, Pufferdaten und einige Programmeinstellungen enthält. Der Dialog des Einlesens eines Projektes enthält, im Vergleich mit einem Standarddialog zum Öffnen eines Fensters, ein Zusatzfenster zur Darstellung von Informationen über die aktuell geöffnete Datei der Projektes. Informationen über das Projekt bestehen aus folgenden Angaben:

- Name und Hersteller des Bauteils
- Datum und Uhrzeit, wann das Projekt gebildet wurde
- Programmversion, in der das Projekt gebildet wurde
- ein durch den Anwender definierter Text über eine nähere Beschreibung des Projektes

Datei / Projekt speichern

Dieser Befehl dient zum Speichern eines Projektes. Der Dialog der Speicherung des Projektes enthält, im Vergleich mit einem Standarddialog zum Öffnen einer Datei, ein Zusatzfenster zur Darstellung von Informationen über das aktuell zu speichernde Projekt.
Es geht um folgende Fenster:

Das dritte Fenster von unten enthält Informationen über die gerade ausgewählte Datei des Projektes im Dialog **Projekt speichern.**

Das zweite Fenster von unten enthält Informationen über das aktuell ausgewählte Bauteil, das Datum, die Uhrzeit und die Version des Programms, in der das Projekt gerade gespeichert wird. Diese Informationen werden in das gerade zu speichernde Projekt hineingelegt.

Das untere Fenster enthält die Beschreibung des Projektes in Form eines Textes, der normalerweise eine durch den Anwender definierte Beschreibung des Projektes, den Namen des Projektautors und einige weitere, durch den Anwender eingetragene Informationen enthält.

Aus den Informationen über das Projekt kann der Anwender nur den letzten Teil direkt editieren – den durch den Anwender eingetragenen beliebigen Text im unteren Fenster des Dialogs **Projekt speichern.** Die Bezeichnung und der Hersteller des Bauteils, das Datum und die Uhrzeit des Projektes und die Programmversion im zweiten und dritten Fenster von unten sind durch das Steuerprogramm automatisch generiert.

Die Informationen und die Beschreibung des Projektes dienen zur näheren Spezifizierung des Projektes, was zur Orientierung des Anwenders bei der Suche nach der gewünschten Datei des Projektes im Dialog **Projekt laden** beitragen soll.

Die Beschreibung des Projektes besteht aus folgenden Bestandteilen:

- Name und Hersteller des aktuell gewählten Bauteils im Projekt
- Datum und Uhrzeit, wann das Projekt gebildet wurde
- Version des Steuerprogramms, in dem das Projekt gebildet wurde
- ein durch den Anwender definierter Text über die nähere Beschreibung des Projektes.

Der Anwender kann nur den letzten Teil direkt editieren – den durch den Anwender eingetragenen beliebigen Text. Die Bezeichnung und der Hersteller des Bauteils, das Datum und die Uhrzeit des Projektes und die Programmversion sind bei jeder Speicherung des Projektes durch das Steuerprogramm automatisch generiert.

Datei / Wiederholtes Laden der Datei

Dieser Befehl ermöglicht, eine Datei, die bereits einmal verwendet wurde, wiederholt einzulesen. Falls Sie eine Datei verwenden, wird der Name dieser Datei auf die Liste der zuletzt verwendeten Dateien im Menü „Wiederholtes Laden der Datei“ eingetragen. Die Dateien sind im Menü der zuletzt verwendeten Dateien nach dem Zeitpunkt der Verwendung so angeordnet, dass die zuletzt verwendeten Dateien vor den früher verwendeten stehen.

Beim erneuten Laden der gewünschten Datei wird wie folgt verfahren:

1. Wählen Sie die Option **Wiederholtes Laden der Datei** aus dem Menü **Datei**.
2. Es erscheint das Submenü, in dem Sie auf das gewünschte Projekt oder auf die gewünschte Datei klicken müssen.

Hinweis: *Sollte eine Datei wiederholt eingelesen werden, ist diese in dem Format eingelesen, in dem die Datei das letzte Mal gelesen oder gespeichert wurde.*

Datei / Wiederholtes Laden des Projektes

Dieser Befehl ermöglicht, ein Projekt, das bereits einmal verwendet wurde, wiederholt einzulesen. Falls Sie ein Projekt verwenden, wird der Name dieses Projektes auf die Liste der zuletzt verwendeten Projekte im Menü **Wiederholtes Laden des Projektes** eingetragen. Die Projekte sind im Menü der zuletzt verwendeten Projekte nach dem Zeitpunkt der Verwendung so angeordnet, dass die zuletzt verwendeten Projekte vor den früher verwendeten stehen

Beim erneuten Laden des gewünschten Projektes aus der Liste wird wie folgt verfahren:

1. Wählen Sie die Option **Wiederholtes Laden des Projektes** aus dem Menü **Datei**.
2. Es erscheint das Submenü, in dem Sie auf das gewünschte Projekt klicken müssen.

Datei / Eigenschaften des Projektes

Dieser Befehl dient zur Darstellung/zum Editieren von Eigenschaften des aktuellen Projektes. Unter Eigenschaften eines Projektes wird die Beschreibung des Projektes verstanden. Die Beschreibung des Projektes besteht aus folgenden Bestandteilen:

- Name und Hersteller des aktuell gewählten Bauteils im Projekt
- Datum und Uhrzeit, wann das Projekt gebildet wurde
- Version des Steuerprogramms, in dem das Projekt gebildet wurde
- ein durch den Anwender definierter Text über die nähere Beschreibung des Projektes.

Der Anwender kann nur den letzten Teil direkt editieren – den durch den Anwender eingetragenen beliebigen Text. Die Bezeichnung und der Hersteller des Bauteils, das Datum und die Uhrzeit des Projektes und die Programmversion sind bei jeder Speicherung des Projektes durch das Steuerprogramm automatisch generiert.

Datei / Verschlüsselungs-Tabelle laden

Dieser Befehl liest die Daten aus der Datei ein, die sich auf der Festplatte im Binärformat befinden und speichert diese im Speicher als Verschlüsselungs-Tabelle (encryption, security,...).

Datei / Verschlüsselungs-Tabelle speichern

Dieser Befehl speichert den Inhalt der Verschlüsselungs-Tabelle (encryption, security,...) in die Datei, die auf der Festplatte im Binärformat gespeichert ist.

Datei / Beenden ohne Speichern

Dieser Befehl beendet das Programm, setzt den zugeordneten Teil des Zentralspeichers frei, löscht den Puffer auf der Festplatte (falls dieser vorhanden war) und übergibt die Steuerung an das Betriebssystem zurück.

Datei / Beenden und Speichern...

Dieser Befehl beendet das Programm, setzt den zugeordneten Teil des Zentralspeichers frei, löscht den Puffer auf der Festplatte (falls dieser vorhanden war) und übergibt die Steuerung an das Betriebssystem zurück.
Der Befehl speichert zusätzlich die aktuelle Konfiguration auf die Festplatte.

Bauteil

Die Funktionen für die Arbeit mit dem gewählten programmierbaren Bauteil – Auswahl des Bauteils, Einlesen des Bauteilinhalts in den Puffer, Leertest, Programmieren und Vergleich der programmierten Daten.

Bauteil / Auswahl aus Default Bauteilen

Dieser Befehl erlaubt es, aus der Liste der Bauteile, die als letzte gewählt wurden, einen Typ des Bauteils auszuwählen. Die Bauteile sind in einem Rundpuffer gespeichert. Aus diesem Rundpuffer können die 10 zuletzt gewählten Bauteile inkl. der Bauteiloptionen wieder aufgerufen werden. Im Puffer können ebenfalls Bauteile gespeichert werden, die sich bereits mit dem Typ der Bauteiloption voneinander unterscheiden. Für die Wahl des Bauteils verwenden Sie bitte die Wahltasten. Diese Bauteilliste wird mit dem Befehl **Datei / Beenden und Speichern** auf die Festplatte gespeichert.

Falls Sie die Wahltaste **<Ctrl+F1>** drücken, bekommen Sie zusätzliche Informationen zum Bauteil, das bereits gewählt wurde. Diese Informationen bestehen aus festgesetzten Daten und aus wahlfreien Daten. Zu den festgesetzten Daten zählen Bauteilgröße, Organisation, Programmieralgorithmus und die Liste der Programmer (einschließlich der gewünschten Module), die dieses Bauteil unterstützen. Die wahlfreien Daten können Gehäuse-Informationen, sowie andere allgemeine Informationen über das jeweilige Bauteil enthalten.

Mit der Wahltaste **<Del>** können Sie ein beliebiges Bauteil aus der Liste der voreingestellten Bauteile löschen. Auf diese Art können Sie zum Beispiel ein Bauteil löschen, das versehentlich in die Liste eingetragen wurde. Es ist nicht möglich, durch das wiederholte Drücken der Taste den Puffer komplett zu löschen. Die letzte Information bleibt im Puffer gespeichert und kann mit dem Drücken der **<Del>** -Taste nicht gelöscht werden.

Bauteil / Bauteilauswahl ...

Dieses Fenster ermöglicht die Auswahl des gewünschten Typs des Bauteils aus der Bibliothek der mit dem aktiven Programmer unterstützten Bauteile durchzuführen. Das Bauteil kann nach dem **Namen**, dem **Typ** oder dem **Hersteller** gewählt werden.

Das ausgewählte Bauteil wird automatisch in den Puffer für die Bauteile übernommen, die zuletzt gewählt worden sind (max. 10 Bauteile). Dieser Puffer kann mit dem Befehl **Bauteil / Auswahl / Default** gewählt werden.

Falls Sie die Wahltaste **<Ctrl+F1>** drücken, bekommen Sie zusätzliche Informationen über das Bauteil, das bereits gewählt wurde. Diese Informationen bestehen aus festgesetzten Daten und aus wahlfreien Daten. Zu den festgesetzten Daten zählen Bauteilgröße, Organisation, Programmieralgorithmus und die Liste

der Programmer (einschließlich der gewünschten Module), die dieses Bauteil unterstützen. Die wahlfreien Daten können Gehäuse-Informationen, sowie andere allgemeine Informationen über das Bauteil enthalten.

Bauteilauswahl ... / Alle

Dieses Fenster ermöglicht es, die Auswahl des gewünschten Bauteiltyps aus der Bibliothek der mit dem aktuellen Programmer unterstützten Bauteile durchzuführen. Die unterstützten Bauteile werden in der Liste angezeigt.

Das Bauteil kann durch Doppelklick auf der Listenzeile mit dem gewünschten Herstellernamen und der gewünschten Bauteilnummer gewählt werden. Das Bauteil kann nicht nur mit den Wahltasten, sondern auch durch direkte Eingabe eines Teiles des Herstellernamens und/oder der Bauteilnummer in die Suchzeile (mit dem Trennzeichen **<Space>**) gewählt werden. Die Auswahl wird mit der **<Enter>**-Taste oder durch das Klicken auf die **OK** Taste bestätigt.

Mit dem Drücken der **<Esc>**-Taste oder durch die Betätigung der **Abbruch**-Taste können Sie jederzeit, ohne dass das aktiv gewählte Bauteil beeinflusst wird, die Auswahl abbrechen.

Das gewählte Bauteil wird automatisch in den Puffer der Bauteile gespeichert, die zuletzt gewählt wurden. Dieser Puffer ist mit dem Befehl **Bauteil / Auswahl / Default** erreichbar.

Mit der Betätigung der **<Ctrl+F1>**-Taste bekommen Sie zusätzliche Informationen über das Bauteil, das bereits gewählt wurde. Diese Informationen bestehen aus festgesetzten Daten und aus wahlfreien Daten. Zu den festgesetzten Daten zählen Bauteilgröße, Organisation, Programmieralgorithmus und die Liste der Programmer (einschließlich der gewünschten Module), die dieses Bauteil unterstützen. Die wahlfreien Daten können Gehäuse-Informationen, sowie andere allgemeine Informationen über das Bauteil enthalten.

Bauteilauswahl ... / Nur ausgewählte Typen

Dieser Befehl ermöglicht die Auswahl des gewünschten Bauteils nach dem Typ durchzuführen. Die mit dem aktiven Programmer unterstützten Bauteile werden in einzelne Gruppen und Untergruppen aufgeteilt. Benutzen Sie die Maus oder die Cursortasten, um vorab die Gruppe (z.B. EPROM) und falls vorhanden, anschließend auch die Untergruppe zu wählen (z.B. EPROM 2Kx8 (2716)). Mit Hilfe des Steuerprogramms wird die Liste der Bauteile, die zu der gewählten Gruppe oder Untergruppe gehören, angezeigt.

Das Bauteil kann ebenfalls durch Doppelklick an der Listenzeile mit gewünschten Herstellernamen und Bauteilnummer gewählt werden. Das Bauteil kann nicht nur mit den Wahltasten, sondern auch durch direkte Eingabe eines Teiles des Herstellernamens und/oder der Bauteilnummer in die Suchzeile (mit dem Trennzeichen **<Space>**) gewählt werden. Die Auswahl wird mit der **<Enter>**-Taste oder durch das Klicken auf die **OK** Taste bestätigt.

Mit dem Drücken der **<Esc>**-Taste oder durch die Betätigung der **Abbruch** -Taste können Sie jederzeit, ohne dass das aktiv gewählte Bauteil beeinflusst wird, die Auswahl abbrechen.

Das gewählte Bauteil wird automatisch in den Puffer der Bauteile gespeichert, die zuletzt gewählt worden sind. Dieser Puffer ist mit dem Befehl **Bauteil / Auswahl / Default** erreichbar.

Mit der Betätigung der**<Ctrl+F1>** -Taste bekommen Sie zusätzliche Informationen über das Bauteil, das bereits gewählt wurde. Diese Informationen bestehen aus festgesetzten Daten und aus wahlfreien Daten. Zu den festgesetzten Daten zählen Bauteilgröße, Organisation, Programmieralgorithmus und die Liste der Programmer (einschließlich der gewünschten Module), die dieses Bauteil unterstützen. Die wahlfreien Daten können Gehäuse-Informationen, sowie andere allgemeine Informationen über das Bauteil enthalten.

Bausteilauswahl ... / Nur ausgewählte Hersteller

Mit diesem Befehl kann der gewünschte Typ des Bauteils nach dem Herstellernamen gewählt werden. Benutzen Sie die Maus oder die Cursortasten, um den Herstellernamen in der Liste der Herstellernamen zu wählen. Mit Hilfe des Steuerprogramms wird die Liste der Bauteile, die zu dem gewählten Herstellernamen gehören, angezeigt.

Das Bauteil kann ebenfalls durch Doppelklick an der Listenzeile mit gewünschten Herstellernamen und Bauteilnummer gewählt werden. Das Bauteil kann nicht nur mit den Wahltasten, sondern auch durch direkte Eingabe eines Teiles des Herstellernamens und/oder der Bauteilnummer in die Suchzeile (mit dem Trennzeichen **<Space>**) gewählt werden. Die Auswahl wird mit der **<Enter>**-Taste oder durch das Klicken auf die **OK** Taste bestätigt.

Mit dem Drücken der **<Esc>**-Taste oder durch die Betätigung der **Abbruch**-Taste können Sie jederzeit, ohne dass das aktiv gewählte Bauteil beeinflusst wird, die Auswahl abbrechen.

Das gewählte Bauteil wird automatisch in den Puffer der Bauteile gespeichert, die zuletzt gewählt worden sind. Dieser Puffer ist mit dem Befehl **Bauteil / Auswahl / Default** erreichbar.

Mit der Betätigung der **<Ctrl+F1>**-Taste bekommen Sie zusätzliche Informationen über das Bauteil, das bereits gewählt wurde. Diese Informationen bestehen aus festgesetzten Daten und aus wahlfreien Daten. Zu den festgesetzten Daten zählen Bauteilgröße, Organisation, Programmieralgorithmus und die Liste der Programmer (einschließlich der gewünschten Module), die dieses Bauteil unterstützen. Die wahlfreien Daten können Gehäuse-Informationen, sowie andere allgemeine Informationen über das Bauteil enthalten.

Bauteil /Auswahl der EPROMs/Flash nach ID

Benutzen Sie diesen Befehl für automatische Auswahl einer EPROM oder Flash durch das Lesen ihrer Identifikationsdaten. Der Programmer kann dabei die Identifikationsdaten des Herstellers und des Bauteils lesen, die in dem EPROM gespeichert sind. Diese Art der Auswahl kann nur bei Bauteilen angewandt werden, die die ID unterstützen. Wird die ID von dem Bauteil nicht unterstützt, wird vom Steuerprogramm eine Warnmeldung angezeigt, dass es sich um ein unbekanntes oder nicht unterstütztes Bauteil handelt.

Werden mehrere EPROMs oder Flash mit identischer ID festgestellt, ist die Liste dieser Bauteile angezeigt. Das gewünschte Bauteil wird aus dieser Liste durch Doppelklick gewählt. Sie können ebenfalls die Wahltasten betätigen und Ihre Wahl durch das Drücken der **<Enter>**- oder der **OK** Taste bestätigen.

Mit dem Drücken der **<Esc>**-Taste oder durch die Betätigung der **Abbruch**-Taste können Sie jederzeit, ohne dass das aktiv gewählte Bauteil beeinflusst wird, die Auswahl abbrechen.

Warnung: *Das Steuerprogramm unterstützt momentan nur die automatische Auswahl einer EPROM mit 28 und 32 Anschlüssen. Für die Programmer, die nicht in der Lage sind, die Anzahl der Bauteilanschlüsse automatisch festzustellen, muss diese Nummer manuell eingegeben werden.*

Der Programmer führt zu den einzelnen Sockelanschlüssen erhöhte Spannung zu, die für die Freigabe des ID-Lesens notwendig ist. Aus diesem Grunde legen Sie niemals in den Sockel andere Bauteile als EPROMs, weil die hohe Spannung an den entsprechenden Anschlüssen das Bauteil beschädigen kann.

Es wird ebenfalls nicht empfohlen, diesen Befehl für EPROMs vom Typ 2764 und 27128 zu benutzen, weil die ID von diesen in meisten Fällen nicht unterstützt werden.

Bauteil / Bauteil Optionen

Dieser Befehl bietet verschiedene Einstellungen, von denen bestimmte Abläufe gesteuert werden. Es handelt sich dabei um den Programmierablauf, Serien-Nummer-Funktion und die Bauteiloptionen.

Bauteil / Bauteil Optionen / Arbeits Optionen

Alle Einstellungen, die dieser Befehl bietet, sind für die Programmierablaufsteuerung bestimmt. Es handelt sich um eine flexible Umgebung, die dem aktiven Bauteil und Programmertyp zugeordnete Begriffe enthält. Begriffe, die zwar für das aktive Bauteil eingesetzt werden können, die aber mit dem aktiven Programmer nicht unterstützt werden, sind ausgeschaltet. Diese Einstellung wird zusammen mit dem aktiven Bauteil auf die Festplatte mit dem Befehl **Datei / Beenden und Speichern** gespeichert. Es handelt sich um folgende Begriffe (die in den

Klammern angeführten Informationen gelten für vorgegebene Werte):

- Gruppe **Adressen**:
  - Bauteil-Start-Adresse (0)
  - Bauteil-End-Adresse (Bauteilgröße-1)
  - Puffer-Start-Adresse (0)
- Gruppe **Einsetzungstest**:
  - Bauteil-Einsetzungstest (ERMÖGLICHEN)
  - Bauteil-ID-Kontrolle (ERMÖGLICHEN)
- Gruppe **Befehlsausführung:**
  - Leertest vor dem Programmieren (AUSGESCHALTET)
  - Bauteillöschen vor dem Programmieren (AUSGESCHALTET)
  - Vergleich nach dem Lesen (AUSGESCHALTET)
  - Vergleich nach dem Programmieren (einmal)
  - Vergleich-Optionen, wenn auf „zweimal“ eingestellt

Anmerkung: *Vergleich nach dem Programmieren kann auf „einmal“ eingestellt werden (bei der Nennspeisespannung) oder auf „zweimal“ (in der Abhängigkeit von der eingestellten Option). Bei dem Vergleich auf „zweimal“ handelt es sich um eine Kontrolle bei niedriger oder bei erhöhter Speisespannung nach Wahl des Anwenders oder nach der Empfehlung des Bauteilherstellers.*

Bauteil / Bauteil Optionen / Serien-Nummer-Funktion

Serien-Nummer-Funktion ist ein spezieller Programmier-modus. Ist die Serien-Nummer-Funktion aktiv, ist der spezifizierte Wert automatisch auf die vordefinierte Adresse in den Puffer bereits vor dem Programmieren jedes Bauteils eingesetzt. Falls mehrere Bauteile nacheinander programmiert werden, wird der Wert der Seriennummer für jedes Bauteil automatisch geändert und vor dem Programmieren des Bauteils in den Puffer eingesetzt, d.h. jedes Bauteil hat eine eigene Seriennummer.

Die Seriennummerfunktion kann zwei Formen haben:

- Inkrement-Modus
- Datei-Modus

Jedes Mal, wenn ein neues Bauteil gewählt wird, wird die Serien-Nummer-Funktion in einen inaktiven Modus eingestellt.
Die aktuellen Einstellungen der Serien-Nummer-Funktion für ein aktiv gewähltes Bauteil werden zusammen mit dem zugeordnetem Bauteil mit dem Befehl **Datei / Beenden und Speichern** auf die Festplatte gespeichert.
Ist der **Inkrement-Modus** aktiv, werden folgende aktuellen Einstellungen gespeichert: Adresse, Größe, Serienwert, Inkrementschritt und die Moduseinstellungen ASCII / BIN, DEC / HEX und LS Byte / MS Byte zuerst.

Ist der **Datei**-Modus aktiv, werden folgende aktuelle Einstellungen gespeichert: Name der Eingangs-Serien-Nummer-Datei und aktuelle Kennzeichnung, mit Hilfe derer die Zeile mit aktueller Serien-Nummer in der Datei angezeigt wird.

Befindet sich das Programm im **Multi-Programm-Modus** (ein Programmer mit mehrfachen Modulen wurde gewählt), so ist im Dialog Serien-Nummer-Funktion die Sektion **Aktion mit (wegen Fehler) nicht programmierten Serien-Nummern** angezeigt. Hier kann man die Operation einstellen, die für Serien-Nummern, welche nicht programmiert wurden, durchgeführt werden soll (am häufigsten wegen eines Fehlers bei Programmieren). Zwei Optionen stehen zur Wahl:

1. die nicht programmierten Serien-Nummern zu ignorieren
2. die nicht programmierten Serien-Nummern in die Datei aufzunehmen

Anmerkungen: *Sollte das Programmieren durch den Anwender unterbrochen werden, ändern sich die Serien-Nummern nicht und sind zum Programmieren bereit. Das Ignorieren, bzw. die Eintragung der nicht programmierten Serien-Nummern in die Datei wird erst dann durchgeführt, wenn die Operation Programmieren (inkl. Vergleich) im vollen Umfang zumindest für ein Bauteil in der gegebenen Bauteilgruppe im Multi-Sockel-Modul abgelaufen ist.*

Die Datei von nicht programmierten Serien-Nummern hat das Text-Format wie die Datei für die Serien-Nummer-Funktion im Datei-Modus. Das heißt, die Serien-Nummern der gegebenen Datei können nachträglich programmiert werden, wobei in den Einstellungen der Serien-Nummer-Funktion der Datei-Modus eingestellt werden muss. Als Bezeichnung der Serien-Nummer-Datei wird die Datei der fehlenden Serien-Nummern gewählt, die nicht programmiert wurden.

Bauteil / Bauteiloptionen / Serien-Nummer-funktion/ Inkrement-Modus

Der Inkrement-Modus ermöglicht es, zu jedem programmierten Bauteil eine individuelle Seriennummer zuzuordnen. Benutzen Sie diese Funktion, mit der die Bauteil-Seriennummer jedes Mal vor dem Einlegen eines neuen Bauteils in den Puffer eingegeben wird. Bei dem Inkrement-Modus wird vom Anwender zuerst die Start-Seriennummer eingegeben, die in den Puffer vor jedem Programmieren gespeichert wird. Nach der Beendigung des Programmierablaufes wird der Wert der Seriennummer um einen spezifischen Schritt inkrementiert und darüber hinaus wird vor jedem Programmierablauf das Speichern einer neuen aktuellen Seriennummer in den Puffer gesichert.

Zur Verfügung stehen folgende Optionen:

Serien-Nummer-Größe
Serien-Nummer-Größe-Option bestimmt die Byte-Anzahl der Serien-Nummer, d.h. die Serien-Nummer-Größe. Für Bin-(binär) Modus ist der Wert von 1 bis 4 Bytes, für ASCII-Modus ist der Wert von 1 bis 8 Bytes zugelassen.

Adresse
Die Adresse-Option bestimmt die Puffer-Adresse, an die Serien-Nummer eingegeben werden soll. Bei Eingabe der Adresse achten Sie bitte darauf, dass die Adresse innerhalb

des Intervalls **Bauteil-Start** und **Bauteil-Ende** liegen muss. Vergessen Sie bitte nicht, dass auch das letzte Byte der Serien-Nummer innerhalb des Intervalls der Adresse **Bauteil-Start** und **Bauteil-Ende** liegen muss.

Startwert
Die Startwert-Option spezifiziert den Initialisationswert (Anfangswert) der Serien-Nummer. Der maximale Wert für eine Serien-Nummer beträgt $1FFFFFFF in einem 32 Bit- Wort. Wird bei dem Inkrementieren der zugelassene Maximalwert überschritten, sind drei Bits der Serien-Nummer mit dem höchsten Wert auf Null abgeglichen und aufgrund dessen gesichert, dass die Nummer wieder innerhalb des Intervalls von 0 bis $1FFFFFFF liegen wird (Überlaufbedienung).

Schritt
Die Schritt-Option bestimmt den Inkrementschritt, um den sich die Serien-Nummer nach Beendigung des Bauteilprogrammierens erhöhen soll. Auf diese Art wird der Wert der Serien-Nummer für das folgende Bauteil gebildet.

Form der Serien-Nummer
Die Form der Serien-Nummer bestimmt die Form, in der die Serien-Nummer in den Puffer eingegeben wird. Es stehen zwei Formen der Serien-Nummer zur Verfügung:

- ASCII
- Bin

ASCII – Serien-Nummer wird in den Puffer in Form der ASCII-Kette eingegeben. Z.B. die Nummer $0528CD wird in Form des ASCII-Modus als 30h 35h 32h 38h 43h 44h (‘0’ ‘5’ ‘2’ ‘8’ ‘C’ ‘D’) in den Puffer eingegeben, das sind 6 Bytes.
Bin – bedeutet, dass die Serien-Nummer direkt in der Form eingegeben wird, in der sie in den Puffer gespeichert wurde. Hat die Serien-Nummer mehr, als ein Byte, kann sie in den Puffer in zwei verschiedenen Formen eingegeben werden (dazu siehe **Speichern im Puffer**).

Schreibart
Die Schreibart-Option bestimmt die Serien-Nummer-Basis, in der die Serien-Nummer dargestellt wird. Es stehen zwei verschiedene Optionen zur Verfügung:

- Dezimal
- Hexadecimal

Dezimal (10) - es werden nur die Dezimalzahlen von 0 bis 9 eingesetzt.
Hexadezimal (16) – es werden Hexadezimalzahlen von A bis F eingesetzt.
Einen Sonderfall stellt der S/N-Modus dar, der auf „Bin“, Schreibart auf „Dec“ eingestellt ist, und das bedeutet BCD-Kodierung. BCD bedeutet, dass die Dezimalnummer in den Hexadezimalnummern gespeichert werden, und für die Vier-Bits, mit denen die einzelnen Zahlen einer Hexadezimalzahl angegeben sind, gilt, dass diese nur die Werte von 0 bis 9

besitzen dürfen (sie dürfen also nicht die Werte von A bis F besitzen).

Speichern im Puffer

Speichern im Puffer ermöglicht, die Art der Eintragung der Serien-Nummer in den Puffer in Abhängigkeit von der Reihenfolge der einzelnen Bytes der Serien-Nummer zu definieren. Die Einstellungen „Speichern im Puffer" können nur für die (Bin)-Form der Serien-Nummer angewandt werden. Es stehen zwei verschiedene Einstellungen zur Verfügung:
Niederwertig - diese Einstellung wird von Intel-Prozessoren benutzt und bedeutet, dass das Byte mit dem niedrigsten Wert sich an der niedrigsten Adresse im Puffer befinden wird.
Höherwertig – diese Einstellung wird von den Prozessoren der Firma Motorola benutzt und bedeutet, dass das Byte mit dem höchsten Wert sich an der niedrigsten Adresse im Puffer befinden wird.

Bauteil / Bauteil Optionen / Serien-Nummer-Funktion / Datei-Modus

Im Falle, dass Datei-Modus verwendet wird, werden die Serien-Werte aus der vom Anwender angegebenen Eingabedatei gelesen und anschließend in den Puffer an die Adresse eingegeben, die in der selben Eingabedatei spezifiziert wird.
Zur Verfügung stehen zwei Datei-Modus-Einstellungen:

Dateiname

Datei-Name-Option spezifiziert den Namen der Eingabedatei, aus der die Serien-Nummer gelesen werden sollen. Datei für den Datei-Modus wird als Text geschrieben. Diese Datei muss das richtige Datei-Format haben. Die Beschreibung des Eingabe-Datei-Formats für den Datei-Modus ist in dem nachfolgenden Abschnitt Datei-Modus **Serien-Nummer-Funktion-Dateiformat** zu finden.

Start-Kennzeichnung

Start-Kennzeichnung definiert die Start-Kennzeichnung in der Eingabedatei. Das Lesen der Serien-Nummer aus der Datei beginnt mit der Kennzeichnung, die in der Start-Kennzeichnung definiert ist.

Datei-modus Serien-Nummer-Funktion-Dateiformat:

Die Eingabedatei der Datei-Modus-Seriennummer-Funktion besitzt ein definiertes Datei-Text-Format, in dem die Byte-Felder definiert sind, von denen die Daten, die für die Eintragung in den Puffer bestimmt sind, angegeben werden. Die Datei für den Datei-Modus wird also von dem Anwender selbst gebildet.

Dateiformat wie folgt:

[label*1*] addr byte*0* byte*1* .. byte*n*
...
[label*n*] addr byte*0* byte*1* . byte*m*, addr byte*0* byte*1* ... byte*k*
_____/ _____/
| |
Grundteil Optionsteil

; Kommentar

Die Bedeutung wie folgt:

Grundteil:
Der Grundteil enthält die Adresse und das Byte-Feld, das in den Puffer eingetragen werden soll. Der Grundteil muss auf jeden Fall in der Zeile nach der Kennzeichnung definiert werden.

Optionsteil:
Der Optionsteil ermöglicht in der entsprechenden Zeile die zweite Adresse sowie das Datenfeld für die Eintragung in den Puffer zu definieren.

label*1*, label*n* - Kennsätze
Die Aufgabe der Kennsätze ist, jede Zeile in der Datei zu identifizieren. Sie werden eingesetzt, um jede Zeile der Eingabedatei adressieren zu können und aus diesem Grund dürfen sich die Bezeichnungen der Kennsätze in der Datei nicht wiederholen. Das Dateizeilenadressieren bedeutet, dass das Lesen der Serien-Nummer mit der Zeile beginnt, in der die Start-Kennzeichnung vom Anwender eingegeben wurde.

addr
Addr definiert die Pufferadresse, an die Daten, die nach der Pufferadresse folgen, eingetragen werden sollen.

byte*0*..byte*n*, byte*0*..byte*m*, byte*0*..byte*k*
Die Byte-Felder byte*0*..byte*n*, byte*0*..byte*m* und byte*0*..byte*k* definieren die Daten, die in den Puffer eingetragen werden sollen. Max. Byte-Anzahl für ein Datenfeld, das nach der Adresse folgt, ist 64 Byte. Datenbytes sind in den Puffer schrittweise, beginnend mit den addr-Adressen wie folgt eingetragen:
byte*0* auf addr
byte*1* auf addr + *1*
byte*2* auf addr + 2
....
byte*n* auf addr + *n*
Optionsteil wird erst nach dem Grundteil angegeben, wobei er von dem Grundteil durch ein Komma getrennt ist. Die Struktur des Optionsteiles ist der Struktur des Grundteiles gleich, d.h. die Adresse und dann folgt das Byte-Feld.

Zeichen mit spezieller Bedeutung:
[ ] – die Kennsätze sollen innerhalb der eckigen Klammern definiert werden,

',' – das Komma dient zur Trennung des Daten-Feldes von dem Grundteil und von dem wahlfreien Bestandteil,

';' – Strichpunkt definiert den Anfang des Kommentars in der Zeile, d.h. alle Zeichen in der Zeile, vom Strichpunkt bis zum

Ende der Zeile werden ignoriert und es folgt die nächste Zeile. Der Kommentar kann sich in einer separaten Zeile befinden oder nach den Datenwerten folgen.

Anmerkungen:

Die Kennsatznamen können beliebige Zeichen beinhalten, nicht aber die Zeichen "[" und "]". Es wird nicht zwischen kleinen und großen Buchstaben unterschieden.

- *Alle Adressen und Byte-Werte werden als Hexadezimalzahlen eingegeben.*
- *Zugelassene Adressengröße in Byte beträgt von 1 bis 4.*
- *Zugelassene Größe der Datenfelder beträgt von 1 bis 64 Bytes, wobei in dem Falle, dass in einer Zeile zwei Datenfelder angegeben werden, darf die Summe der Datenfelder max. 80 Bytes betragen.*
- *Seien Sie vorsichtig bei der Eingabe der Adressen. Die Adressen sollen innerhalb des Intervalls* ***Bauteil-Start*** *und* ***Bauteil-Ende*** *liegen. Liegt irgendeine Adresse außerhalb dieses Intervalls, wird ein Fenster mit Hinweis auf diesen Zustand angezeigt und die Serien-Nummer-Funktion wird in den inaktiven Modus umgeschaltet (inaktiver Modus).*

Beispiel für eine typische Eingabedatei für den Datei-Modus:

```
[nav1] A7890 78 89 56 02 AB CD ; Kommentar1
[nav2] A7890 02 02 04 06 08 0A
[nav3] A7890 08 09 0A 0B A0 C0 ; Kommentar2
[nav4] A7890 68 87 50 02 0B 8D
[nav5] A7890 A8 88 59 02 AB 7D

;die folgende Zeile enthält ebenfalls die zweite Definition:
[nav6] A7890 18 29 36 42 5B 6D, FFFF6 44 11 22 33 99 88 77
66 55 16

;das ist der Kommentar: die letzte Zeile – Dateiende.
```

In dieser Datei, die als Beispiel angeführt wurde, sind sechs Serien-Nummer mit den Kennsätzen „nav*1*", „nav*2*", ... „nav*6*" definiert. Jede von diesen Nummern ist in dem Puffer an der Adresse $A7890 eingetragen. Die Größe von allen Werten beträgt sechs Bytes. Die Zeile mit dem Kennsatz „nav*6*" beinhaltet zusätzlich die Definition des zweiten Wertes, der z.B. im CRC-Falle die Summe darstellen kann, und der zweite Wert ist in dem Puffer an der Adresse $FFFF6 gespeichert. Die Größe des Wertes beträgt 10 Bytes, d.h. das letzte Byte dieses Wertes wird in den Puffer an die Adresse $FFFFF eingeschrieben.

Bauteil / Bauteil Optionen / Statistik

Die Statistik gibt die Information über die aktuelle Zahl der Bauteiloperationen, die mit dem gewählten Bauteil durchgeführt wurden. Falls einem Bauteil eine Operation entspricht, z.B. das

Programmieren, ist in dem Falle die Zahl der Operationen der Zahl der programmierten Bauteile gleich.

Die nächste Funktion der Statistik ist die Einstellung bestimmter Zahl der Bauteile, an denen bestimmte Operation durchgeführt werden sollen, wobei nach jeder erfolgreich durchgeführten Operation ein sog. **Abwärtszählen** oder ein anderer Zähler dekrementiert wird, der zu Beginn die von dem Anwender angegebene Anzahl der Bauteile beinhaltet. Wird vom **Abwärtszählen** die Null erreicht, bedeutet es, die Operationen einer bestimmten Anzahl der Bauteile bereits durchgeführt wurden, und für den Anwender wird darüber eine Information angezeigt. Die Frage über Wiederholung der Operation wird nicht mehr angezeigt.

Dialog **Statistik** enthält folgende Optionen:

- Die Checkbox **Programmieren**, **Vergleich**, **Leertest**, **Löschen** und **Lesen** definiert die Operationen, nach denen die Statistikwerte inkrementiert werden sollen.

- Die Checkbox **Abwärtszählen** bestimmt, ob das Abzählen aktiv oder inaktiv ist. In das Edit-Fenster nach der Checkbox **Abwärtszählen** wird die Ausgangzahl der Bauteile angegeben. Das Abzählen fängt mit diesem Wert an.

Befindet sich der Cursor auf dem **Statistik-Feld**, besteht die Möglichkeit, dieses Feld mit dem Klicken der rechten Maustaste und durch die Bestätigung der **Statistik**-Option zu öffnen.

Die aktuellen statistischen Werte werden im Hauptfenster des Steuerprogramms im **Statistik**-Feld angezeigt.

Statistik-Feld enthält drei Werte von statistischen Kenngrößen: **Erfolg**, **Fehler, Total** und zwei Werte, die Sie über den Stand des Abzählens informieren:
Abwärtszählen: und **Rest:**.

Bedeutung dieser Werte wie folgt:

Erfolg	gibt die Zahl der Operationen, die erfolgreich beendet wurden.
Fehler	gibt die Zahl der Operationen, die erfolglos beendet wurden.
Total	gibt die Gesamtzahl der Operationen.
Abwärtszählen	gibt die Information darüber, ob das Abwärtszählen aktiv oder inaktiv ist.
Rest	gibt die Information darüber, wie viel Operationen noch durchzuführen sind.

Unter einer erfolgreichen Operation versteht man die erfolgreiche Durchführung einer von nachfolgend angeführten Operationen:

- Programmieren
- Vergleich

- Leertest
- Löschen
- Lesen

Ist eine von diesen Operationen mit einem Fehler durchgeführt worden, wird diese Operation als eine nicht erfolgreiche Operation betrachtet.

Sollte ein neues Bauteil angewählt werden, werden in dem Falle alle statistischen Werte auf Null und das **Abwärtszählen** auf **Deaktiviert** eingestellt.
Reset-Taste im **Statistik-**Feld löscht die statistischen Werte.
Abwärts-Zählen-Setzen-Taste im **Statistik-**Feld setzt den Abwärtszählen auf die Bauteil-Ausgangszahl zurück.

Bauteil / Bauteil Optionen / Zugeordnete Datei

Dieser Befehl ermöglicht, den vollständigen Dateinamen, der mit dem aktuellen Bauteil zusammenhängt, von Hand anzugeben. Es handelt sich dabei um eine Datei, die bei der Bauteil-Auswahl aus den vorgegebenen Bauteilen automatisch (d.h. **<F5>** Option), oder bei dem Programm-Start nach der Bearbeitung der sich auf der Festplatte befindenden Datei in den Puffer eingelesen werden kann. Der Dateiname kann in jedem Fall bearbeitet werden, d.h. auch in dem Falle, dass das automatische Lesen auf diese Art eingegebener Datei ausgeschaltet ist. Die zugeordnete Datei sowie die Freigabe des automatischen Lesens der Datei bei der Bauteil-Auswahl wird auf die Festplatte mit dem Befehl **Datei / Beenden und Speichern** gespeichert.

Durch Auswahl eines neuen Bauteils wird das neue Bauteil auf den leeren Dateinamen voreingestellt und das automatische Lesen wird ausgeschaltet.

Bauteil / Leertest

Dieser Befehl ermöglicht es, den Leertest des ganzen Bauteils (oder eines Teiles des Bauteils) durchzuführen. Das Resultat wird vom Steuerprogramm durch entsprechende Meldung, die auf dem Bildschirm erscheint, angezeigt.

Sollte der Leertest nur bei einem Teil des Bauteils durchgeführt werden, wird im Menü **Bauteil / Bauteil Optionen/ Arbeits Optionen** die Eingabe- und Endadresse des aktuellen Bereiches unter der Voraussetzung eingestellt, dass für das entsprechende Bauteil diese Option unterstützt wird.

Bauteil / Lesen

Dieser Befehl ermöglicht, das ganze Bauteil (oder einen Teil des Bauteils) zu lesen und dieses dann in den Puffer des Steuerprogramms zu speichern. Das Resultat wird vom Steuerprogramm durch entsprechende Meldung, die auf dem Bildschirm erscheint, gemeldet.

Sollte das Lesen nur bei einem Teil des Bauteils durchgeführt werden, wird im Menü **Bauteil / Bauteil Optionen / Arbeits**

Optionen die Eingabe- und Endadresse des aktuellen Arbeitsbereiches unter der Voraussetzung eingestellt werden, dass für das entsprechende Bauteil diese Option unterstützt wird. Ebenfalls kann die die automatische Option **Vergleich nach dem Lesen** eingestellt werden, durch die Zuverlässigkeit des Bauteils-Lesens erhöht wird.

Bauteil / Vergleich

Dieser Befehl vergleicht die programmierten Daten des Bauteils mit den im Puffer des Steuerprogramms gespeicherten Daten im gültigen Arbeitsbereich der voreingestellten Adressen. Das Resultat wird vom Steuerprogramm durch entsprechende Meldung, die auf dem Bildschirm erscheint, angezeigt.

Der gültige Arbeitsbereich wird im Menü **Bauteil / Bauteil Optionen / Arbeits Optionen** unter der Voraussetzung eingestellt, dass für das entsprechende Bauteil diese Option unterstützt wird.

Stellen Sie im Menü **Optionen / Fehler anzeigen** die Art, auf die aufgetretene Fehler angezeigt werden sollen. Die aufgetretenen Fehler können in die Datei VERIFY.ERR eingeschrieben werden oder auch auf dem Bildschirm angezeigt werden. Sollten die Fehler auf dem Bildschirm angezeigt werden, können nur die ersten 45 aufgetretenen Fehler angezeigt werden.

Bauteil / Programmieren

Dieser Befehl ermöglicht das ganze Bauteil (oder ein Teil des Bauteils) mit den Daten aus dem Puffer zu programmieren. Das Resultat wird vom Steuerprogramm durch entsprechende Meldung, die auf dem Bildschirm erscheint, angezeigt.

Die Steuerung des Programmierablaufes erfolgt mit Hilfe der Einstellungen im Menü **Bauteil / Bauteil Optionen / Arbeits Optionen**. Es handelt sich dabei um eine variable Umgebung, die einstellbare, mit dem aktuellen Bauteil und mit dem Typ des Programmers zusammenhängende Daten enthält.

Bauteil / Löschen

Dieser Befehl ermöglicht, das ganze programmierbare Bauteil zu löschen. Das Resultat wird vom Steuerprogramm durch entsprechende Meldung, die auf dem Bildschirm erscheint, angezeigt.

Bauteil / Test

Dieser Befehl ermöglicht, den Test eines aus der Bibliothek der unterstützenden Bauteile ausgewählten Bauteils durchzuführen (z.B. Test der statischen RAMs), soweit dieser Test mit dem aktuellen Programmer unterstützt wird.

Bauteil / Bauteilinfo

Dieser Befehl bietet zusätzliche Informationen über das aktuelle Bauteil. Diese Informationen bestehen aus festgesetzten und aus wahlfreien Informationen. Die festgesetzten Informationen enthalten Bauteilgröße, Organisation, Programmieralgorithmus und Programmerauflistung (inkl. der benötigten Module), von denen dieses Bauteil unterstützt wird. Die wahlfreien Informationen können Informationen über das Gehäuse sowie verschiedene allgemeinen Informationen über das Bauteil enthalten.

Um diesen Befehl zu aktivieren, verwenden Sie die **<Ctrl+F1>** - Wahltaste.

Puffer

Das Puffer-Submenü bietet breite Möglichkeiten für die Arbeit mit dem Puffer, wie z.B. Block-Operationen, Inhaltlöschen, Ausfüllung eines Teiles des Puffers mit der Zeichenkette, 4 verschiedene Prüfsummen und natürlich auch das Editieren, das weitere zusätzlichen Operationen anbieten kann (Zeichenketten-Suchung, Ersatz der Zeichenketten, Output für den Drucker usw.).

Puffer / Anzeigen/Editieren...

Dieser Befehl wird für das **Anzeigen** (Anzeige-Modus) oder für das **Editieren** (Editier-modus) der Daten im Puffer benutzt. Benutzen Sie die Pfeiltasten oder Maustasten, um sich im gesamten Puffer bewegen und der gewünschte Block für das Bearbeiten wählen zu können. Modifizierte Daten unterscheiden sich in ihrer Farbcodierung.

Ebenfalls können Sie die **<F4>** Taste verwenden.

Anzeigen / Editieren Puffer

F1	Hilfe Anzeige der Hilfe zum aktuellen Fenster
F2	Pufferblock auffüllen ermöglicht den gewählten Pufferblock mit gewünschter Zeichenkette aufzufüllen. Diese kann entweder als ASCII- oder als HEX-Zeichenkette eingegeben werden. Maximallänge der Kette beträgt 16 ASCII-Zeichen.
Ctrl+F2	Puffer löschen - dieser Befehl löscht den Puffer mit bestimmten Leerzeichen (für EPROM ist es 0FFH).
Ctrl+Shift+F2	füllt den Puffer mit Zufall-Daten auf.
F3	Pufferblock kopieren kopiert den gewünschten Pufferblock an die neue Adresse. Die Zieladresse muss aber außerhalb der Quelladresse liegen.
F4	Pufferblock-Übertragung überträgt den Quellpufferblock in den Zielpufferblock, d.h. an eine neue Adresse. Die Zieladresse muss aber außerhalb der Quelladresse liegen. Der Quellpufferblock (oder ein Teil des Quellpufferblocks) wird nach der Pufferblock-Übertragung mit bestimmten Leerzeichen aufgefüllt (für EPROM ist es 0FFH).
F5	Block umtauschen: der Befehl tauscht gegenseitig die Byte-Paarungen im gewählten Pufferblock um. Es wird vorausgesetzt, dass der Block an einer geraden Adresse startet und das er auch eine gerade Anzahl der Bytes enthält. Sind diese Voraussetzungen nicht erfüllt, werden die Adressen vom Programm selbst modifiziert (d.h. die Startadresse wird an die untere gerade Adresse und/oder die Endadresse an die höhere gerade Adresse

	verschoben)
F6	Drucken Puffer drucken
F7	Suchen der Befehl löst das Suchen der Zeichenkette aus (Maximallänge der Zeichenkette: 16 ASCII -Zeichen).
F8	Ersetzen die Zeichenkette wird gefunden und ersetzt (max. 16 ASCII Zeichen).
F9	Sprung auf die Adresse ändert die aktuelle Adresse
F10	ändert den Durchsuchungs-Modus / Bearbeitungs-Modus
F11	schaltet den Anzeige-Modus von Pufferdaten zwischen 8 Bits- und 16 Bit-Darstellung um. Die Umschaltung kann auch mit Mausklicken auf die Taste neben dem Durchsuchensmodus-Indikator oder neben dem Puffer-Bearbeitungs-Modus erfolgen. Diese Taste zeigt gleichzeitig den aktuellen Darstellungs-Modus an. (8 Bits oder 16 Bits).
F12	Dialog für die Berechnung der Prüfsumme ermöglicht, die Prüfsummen in dem gewählten Datenblock im Puffer zu berechnen.
Esc	Schließen der Befehl beendet die Pufferverarbeitung und durchsuchung.
Pfeiltasten	Cursor oben, unten, rechts und links
Home/End	Sprung auf den Start / Ende der aktuellen Zeile
PgUp/PgDn	Sprung auf vorherige /nächste Seite
Ctrl+PgUp/PgDn	Sprung auf den Start / Ende des aktuellen Puffers
Ctrl+Home/End	Sprung auf den Start / Ende der aktuellen Seite
Backspace	verschiebt den Cursor um eine Position nach links (zurück)

Anmerkung: *Die Zeichen 20H - FFH (**ASCII-Regime**) und die Zahlen 0..9, A..F (**HEX-Regime**) ändern unmittelbar den Inhalt des zu bearbeitenden Puffers.*

Puffer Drucken

Dieser Befehl ermöglicht, den Inhalt des gewählten Pufferblocks mit dem Drucker zu drucken oder diesen in eine Datei auf die Festplatte zu schreiben. Das Programm benutzt dabei einen externen Texteditor, der vom Anwender gewählt wurde. Mit Hilfe dieses Texteditors wird der Inhalt des Pufferblocks angezeigt und er kann auch gedruckt werden. Normalerweise ist der **Notepad.exe**-Texteditor voreingestellt, der auch der Bestandteil aller Versionen des MS Windows-Systems ist.

Im Dialog **Puffer drucken** stehen folgende Einstellungen zur Verfügung:

Block Start
Definiert die Startadresse des Pufferblocks, der gedruckt (dargestellt) werden soll. Ist der Puffer als Byte-Puffer dargestellt,

werden Bytes adressiert; ist der Puffer als 16 Bit-Puffer dargestellt, werden 16 Bit-Wörter adressiert.

Block Ende
Definiert die Endadresse des Pufferblocks, der gedruckt (dargestellt) werden sollte. Ist der Puffer als Byte-Puffer dargestellt, werden Bytes adressiert; ist der Puffer als 16 Bit-Puffer dargestellt, werden 16 Bit-Wörter adressiert.

Externer Editor
Definiert den Pfad sowie den Name des externen Programms, der als Texteditor für ausgewählten Pufferblock eingesetzt werden sollte. Normalerweise ist der Notepad.exe-Texteditor voreingestellt, der auch der Bestandteil aller Versionen des MS Windows-Systems ist.
Vom Anwender kann auch ein beliebiger Texteditor definiert werden, z.B. Wordpad.exe, der in der Lage ist, ebenfalls große Textdateien zu verarbeiten. Der Anwender kann den Puffer im ausgewählten Texteditor drucken.

Die Pfadeinstellung sowie der Name des Texteditors werden nach der Beendigung des Steuerprogramms automatisch auf die Festplatte gespeichert.

Dialog box suchen

Dieses Fenster dient zum **Suchen** der Zeichenkette im Puffer. In **Zeichenkette suchen** wird die Zeichenkette eingegeben, die gesucht wird, und nach Bestätigung beginnt das Suchen. Nach Betätigung der Taste **Abbruch** wird das Fenster **Dialog Box Suchen** geschlossen.

Such-Richtungs-Box definiert die Richtung des Suchens (vorwärts oder rückwärts). Als Bezugspunkt dient die aktuelle Cursor-Position im Durchsuchungsmodus. Im Bearbeitungsmodus wird der gesamte Puffer durchgesucht.

Anfang spezifiziert, wo das Suchen starten soll (von der Cursor-Position oder im ganzen Puffer).

Dialog box ersetzen

Mit diesem Befehl kann die gefundene Zeichenkette durch eine andere Zeichenkette ersetzt werden. Unter **Dialog Box suchen** geben Sie die Zeichenkette ein, die Sie ersetzen wollen und unter **Ersetzen** geben Sie die Zeichenkette ein, die ursprüngliche Zeichenkette ersetzen soll.

Klicken Sie auf **Einstellungen** und geben Sie an, ob Programm die Bestätigung zum Ersatz der Zeichenkette anfragen soll oder nicht.

Anfang spezifiziert, wo das Suchen starten soll (von der Cursor-Position oder im ganzen Puffer).

Such-Richtungs-Box definiert die Richtung des Suchens (vorwärts oder rückwärts). Als Bezugspunkt dient die aktuelle Cursor-Position im Editmodus. Im View-Modus wird der gesamte Puffer durchgesucht.

Betätigen Sie die **<Esc>**-Taste oder klicken Sie auf die Taste **<Cancel>**, falls Sie das Fenster **Dialog Box ersetzen** schließen wollen.

Falls Sie auf die Taste <**Ersetzen>** klicken, schließt sich das Fenster **Dialog Box ersetzen** und es wird ein **Frage-**Fenster angezeigt. Im Frage-Fenster stehen folgende Optionen zur Verfügung:

Ja	ersetzt die gefundene Zeichenfolge und sucht nach der nächsten.
Nein	sucht die nächste Zeichenfolge
Alles ersetzen	ersetzt alle gefundenen Zeichenfolgen
Suche abbrechen	beendet das Suchen sowie den Zeichenfolgen-Ersatz.

Puffer Anzeigen / Puffer für PLD Editieren

Ctrl+F2	Pufferlöschen mit voreingestelltem Leerzeichen.
Ctrl+Shift+F2	Auffüllen des Puffers mit Zufall-Daten.
F9	Sprung auf die Adresse
F10	Anzeige/Editier-Modus ändern.
F11	Umschalten der Pufferdaten zwischen 1 Bit und 8 Bit Ansicht. Die Umschaltung kann auch mit Mausklicken auf die Taste neben dem Durchsuchensmodus-Indikator oder neben dem Puffer-Bearbeitungs-Modus erfolgen. Diese Taste zeigt gleichzeitig den aktuellen Darstellungs-Modus an. (1 Bits oder 8 Bits).
Pfeiltasten	schieben den Cursor nach oben, unten, rechts und links.
Home/End	Sprung auf Start /Ende der aktuellen Zeile.
PgUp/PgDn	Sprung auf die vorherige / nächste Seite.
Ctrl+PgUp/PgDn	Sprung auf den Start /Ende des Puffers.
Ctrl+Home/End	Sprung auf Anfang /Ende der Seite.
Backspace	bringt den Cursor um eine Position nach links zurück.

Anmerkung: *Die Zeichen 0 und 1 ändern direkt den Inhalt des Bearbeitungsbereiches*.

Puffer / Block füllen

Dieser Befehl ermöglicht, den gewählten Block des Puffers mit der vorgegebenen Zeichenkette auszufüllen. Die Zeichenkette kann entweder im ASCII-Format oder im HEX-Format eingegeben werden. Maximale Länge der Zeichenkette beträgt 16 ASCII-Zeichen.

Puffer / Block kopieren

Mit diesem Befehl können Sie den bezeichneten Block der Daten im aktuellen Puffer an die neue Adresse kopieren. Die Zieladresse, unter der die Daten gespeichert werden, darf nicht mit der Ausgangsadresse identisch sein.

Puffer / Block verschieben

Mit diesem Befehl wird der Quellenblock der Daten in dem aktuellen Puffer an die neue Adresse, d.h. in den Zielpuffer übertragen. Die Quellenblockadresse darf nicht mit der Zielblockadresse identisch sein. Der Quellenblock (oder ein Teil des Quellenblocks) wird nach der Übertragung der Daten mit aktuellen Leerzeichen-Konstante aufgefüllt (für EPROM ist es 0FFH).

Puffer / Block vertauschen

Der Befehl tauscht gegenseitig Byte-Paare im aktuellen Pufferblock um. Es wird vorausgesetzt, dass der Block an der geraden Adresse startet und dass er eine gerade Byte-Zahl enthält. Sind die Voraussetzungen nicht erfüllt, werden die Adressen vom Steuerprogramm selbst modifiziert (die Startadresse wird an die untere gerade Adresse und /oder die Endadresse an die höhere ungerade Adresse verschoben).

Puffer / Löschen

Ist dieser Befehl gewählt, wird der Inhalt des Puffers mit aktuellem Leerzeichen aufgefüllt (für EPROM ist es 0FFH).

Ebenfalls können Sie die **<Ctrl+F2>** Taste verwenden.

Puffer / Füllen mit zufälligen Daten

Mit diesem Befehl wird der Puffer mit Zufälligen Daten aufgefüllt.

Für diesen Befehl ist die**<Shift+Ctrl+F2>**-Taste reserviert.

Puffer / Puffer-Inhalt duplizieren

Dieser Befehl ermöglicht, den Pufferinhalt aus dem Bereich des EPROM-Quellengedächtnisses in den EPROM-Zielbereich zu duplizieren. Das Duplizieren des Puffers ist geeignet, wenn das EPROM 27C512-Gedächtnis anstatt des EPROM 27C256-Gedächtnisses verwendet werden soll.

Anmerkung: *Zum Duplizieren des Puffers wird immer die Adresse des Pufferanfangs (gleich Null) verwendet.*

Puffer / Prüfsumme

Nach Eingabe der Pufferadressen in der Hexadezimalform und nach ihrer Bestätigung wird die Prüfsumme auf folgende Art berechnet:

Byte	Summierung nach einzelnen Bytes ins "word". CY-Merkmal ist ignoriert.
Word	Summierung nach einzelnen Wörtern ins "word". CY-Merkmal ist ignoriert.
Byte (CY)	Summierung nach einzelnen Bytes ins

	"word". CY-Merkmal wird zum Resultat zugerechnet.
Word (CY)	Summierung nach einzelnen Wörtern ins "word". CY-Merkmal wird zum Resultat zugerechnet.
CRC-CCITT	Summierung nach einzelnen Bytes ins "word" , wobei folgende Formel gilt: RESULTAT=VORHERIGES+ (x^16 + x^12 + x^5 +1)
CRC-XModem	Summierung nach einzelnen Bytes ins "word" , wobei folgende Formel gilt: RESULTAT=VORHERIGES + (x^16 + x^15 + x^2 +1)

Kolonne, die als **Neg.** bezeichnet ist**,** stellt die Negation der Prüfsumme dar, d.h. es gilt: SUM + NEG. = FFFFH.
Kolonne, die als **Komplement** bezeichnet ist, ist das Gegenstück der Prüfsumme, es gilt also: SUM + SUPPL. = 0 (+ Übertragung).

Prüfsumme-Dialog enthält folgende Felder:

Von der Adresse: die Startadresse des gewählten Pufferblocks für die Berechnung der Prüfsumme im Puffer. Die Adresse ist als Byte-Adresse definiert.

Bis zur Adresse: die Endadresse des gewählten Pufferblocks für die Berechnung der Prüfsumme im Puffer. Die Adresse ist als Byte-Adresse definiert.

Prüfsumme einfügen: Auswahl der Art der Prüfsumme, die nach Durchführung des Befehls **Berechnen und Einfügen** in den Puffer eingeschrieben wird.

Einfügen in die Adresse: Pufferadresse, in die Prüfsumme nach Durchführung des Befehls **Berechnen & Einfügen** eingeschrieben wird**.** Die Adresse kann aber nicht innerhalb des Bereiches <**Von der Adresse**> <**Bis zur Adresse**> liegen. Die Adresse ist als Byte-Adresse definiert.

Größe: Einstellung der Resultat-Größe der gewählten Prüfsumme. Die Resultat-Größe der Prüfsumme kann 8 (Byte) oder 16 (Word) Bytes betragen. Ist die 16 (Word) Byte-Größe gewählt, wird die gesamte Prüfsumme in den Puffer eingeschrieben, sonst wird in den Puffer nur das untere Byte der Prüfsumme eingeschrieben werden.

Anmerkung: *Falls 16 (Word)-Größe gewählt wird, wird das untere Byte der Prüfsumme in die Adresse, die im Feld* ***Einfügen an Adresse*** *spezifiziert ist und das obere Byte in die Adresse, die um eine Adresse höher liegt, eingeschrieben.*

Optionen

Im Menü sind Befehle enthalten, mit deren Hilfe verschiedene Programm-Voreinstellungen angezeigt oder geändert werden können.

Optionen / Generelle Optionen

Im Dialog Generelle Optionen können folgende allgemeine Programmeigenschaften eingestellt werden

Datei-Optionen

Datei-Optionen-Seite enthält Datei-Suffix-Einstellungen in einzelnen Formaten, die Art des wiederholten Lesens der aktuellen Datei während der Modifikation der Datei durch andere Applikationen sowie die Format-Erkennung der eingelesenen Dateien.

Datei-Format-Masken – in den Editierfenstern **Datei-Format-Masken** werden Dateimasken eingestellt, die als Filter für angezeigte Dateiliste im Speicher-Fenster **Datei/ Speichern** und Lesen **Datei / Laden** dienen (jedes Format verfügt über eigene Maske für Speichern). Die Maske muss mindestens eins von verwendeten Jockerzeichen (*,?) enthalten und außerdem muss sie syntaktisch richtig eingegeben werden, sonst kann sie nicht akzeptiert werden.

Das Suffix der Projekt-Dateien ermöglicht es, das Suffix für Projekte einzustellen, die das Steuerprogramm verwendet. Das Suffix wird in den Dialogen **Datei / Laden** und **Datei/ Speichern** verwendet.

Ist die aktuelle Datei durch andere Applikation modifiziert, kann die Art des wiederholten Lesens der aktuellen Datei eingestellt werden, falls die Datei durch eine andere Applikation geändert wurde. Es stehen drei Möglichkeiten zur Verfügung.

Das Format der bereits eingelesenen Datei steuert das Einlesen der gewählten Datei von der Festplatte in den Puffer mit einer **automatischen** oder **manuellen** (vom Anwender angegeben) Format-Erkennung. Wird die Datei von der Festplatte in den Puffer mit **automatischer** Format-Erkennung eingelesen, versucht das Programm, das Format der zu lesenden Datei zu analysieren, wobei von ihm alle, durch das Programm unterstützten Formate getestet werden. Verfügt die Datei über eins der vorgegebenen Formate, wird die Datei auf entsprechende Weise in den Puffer eingelesen.
Die **manuelle** Format-Auswahl für das Einlesen der Datei von der Festplatte in den Puffer ermöglicht dem Anwender, im Format-Auswahl-Fenster das entsprechende Format nach Wunsch des Anwenders auszuwählen. Verfügt die Datei aber nicht über das vom Anwender ausgewählte Format, kann natürlich passieren, dass das Einlesen der Datei erfolglos wird. Das gilt insbesondere für die Formate in der ASCII-Form mit Normformaten, die aufgrund Standards gegeben sind.

Optionen für Hex-Dateien

Im Puffer erfolgt das Einlesen der externen Datei in den Puffer. Dieser Befehl ermöglicht, die Dateidaten-Lesen-Steuerung auf beliebige Art und mit beliebigem Format durchzuführen.

Die erste Option stellt das **automatische Löschen des Puffers vor dem Lesen** mit der Leerzeilen-Konstante ein (z.B. für EPROM ist es 0FFH).

Die zweite Option stellt einen **negativen Offset** ein, der für die Modifikation der aus der Datei gelesenen Daten-Adressen dient und zwar auf die Art, dass sie ihre Eintragung an die vorhandenen Pufferadressen ermöglicht.

Beispiel:

Die Datei enthält Daten im S – Format der Firma Motorola. Der Datenblock startet an der FFFF0H-Adresse. Es handelt sich dabei um ein S2-Format, d.h. der Zeileneintrag enthält ein Adressenfeld mit der Größe von 3 Bytes. Um alle Daten lesen zu können, müssen diese in den Puffer beginnend mit der Adresse 0 gespeichert werden. Offset wird auf den FFFF0H-Wert eingestellt und darüber hinaus wird gesichert, dass von den aktuellen Daten-Adressen, die in der Datei vorhanden sind, bereits dieser Wert abgerechnet wird und dass die Eintragung dieser Daten bei der Null-Adresse beginnt.

Warnung: *Weil der Wert des eingestellten Offsets von der realen Adresse abgezogen wird, beachten Sie die Richtigkeit der Adresseneingabe. Sonst kann es passieren, dass die modifizierte Adresse einen negativen (und darüber hinaus auch einen falschen) Wert besitzen wird. Der tatsächliche, aus der Datei gelesene Adressenbereich ist mit* ***Datei-Anfang*** *und* ***Datei-Ende*** *des Hauptfensters des Steuerprogramms gegeben.*

Die dritte Option ist die **Umadressierung** der Daten aus dem **Intel-extended-HEX-Format** an die vorhandenen Pufferadressen. Das bedeutet, dass von dem Anwender das Segment eingegeben wird, das dem niedrigsten Segment der Datei zugeordnet wird (02-Aufnahme-Typ) und dass die anderen Segmente mit dieser Differenz modifiziert werden. Diese Option ist aber nicht für das „einfache" Intel- HEX-Format zugelassen. Standardoption ist im **Standardmodus** eingestellt.

Beispiel:
Die Datei enthält zwei Records, Typ 02, mit Adressen F000H und F800H. Falls ein neues Segment, z.B. 0H-Segment, eingegeben wird, werden die Daten vom F000H-Segment in das 0H-Segment umgeleitet. Dasselbe gilt auch umgekehrt, also die Daten vom F800H-Segment werden in das 800H-Segment umgeleitet.

Diese Einstellung wird in die Konfigurationsdatei gespeichert, falls die Option **Optionen speichern** gewählt wird, oder bei der Beendigung des Programms, falls die Speicherung der Konfiguration eingestellt wurde.

Sprache

Diese Option erlaubt, verschiedene Sprachen für das Anwender-Interface einzustellen, wie z.B. Menü, Tasten, Dialoge, Informationen und Fehlermeldungen.
Außerdem ermöglicht sie, die Hilfedatei in verschiedenen Sprachen anzuwählen. Für die Unterstützung der konkreten Sprache werden Sie entsprechende externe Sprache-Definitions-Dateien in der Text-Form brauchen.

Sound

Die Seite **Sound** erlaubt dem Anwender, das Programm-Sound-Modus zu wählen. Das Programm generiert Sound nach Beendigung bestimmter Aktivitäten, z.B. nach den Aktivitäten mit dem Bauteil (Programmieren, Vergleich, usw.). Vom Programm wird der Sound auch in dem Falle generiert, dass eine Warnung oder Fehlermeldung angezeigt wird.
Der Anwender kann zwischen folgenden Sound-Generier-Arten wählen:

1. MS Windows-Sound (installierte Soundkarte erforderlich)
2. Langer Ton über einen PC-System-Lautsprecher
3. Kurzer Ton über einen PC-System-Lautsprecher
4. Kein Sound

Andere

Auf der Seite **Andere** können verschiedene Programmeigenschaften eingestellt werden. Auf der **Tastenleiste** kann entweder die flache Tastenanzeige sowie die Blasenhilfe für die Tasten im Programmhauptfenster eingestellt werden. Auf der Leiste **Startinhaltverzeichnis** kann die Art für die Startverzeichnisbestimmung beim Start des Programms eingestellt werden. Die Option **Voreingestelltes Inhaltverzeichnis** bedeutet, dass als Startverzeichnis des Programms das Inhaltverzeichnis bezeichnet wird, aus dem das Programm nach dem Start aufgerufen wird. Die Option **Inhaltverzeichnis, mit dem das Programm zuletzt beendet wurde,** bedeutet, dass als das Startverzeichnis des Programms bei seinem Start das Inhaltverzeichnis eingestellt wird, auf das das Programm bei der letzten Programmbeendigung eingestellt wurde. Im Grunde genommen handelt es sich hier um das erste (aktuellste) Inhaltverzeichnis aus der Inhaltverzeichnisliste.

Optionen Speichern

Diese Option ermöglicht die Art des Speicherns von aktuellen Einstellungen bei der Programmbeendigung zu wählen. Es stehen drei Möglichkeiten zur Verfügung:

Optionen nicht speichern – d.h. die Optionen nach der Programmbeendigung nicht zu speichern und die Option **Speichern** nicht anzufragen.

Optionen automatisch speichern – d.h. die Optionen automatisch, ohne dass die Option **Speichern** angefragt wird, bei der Programmbeendigung zu speichern.

Aufforderung vor der Speicherung der Optionen – bei der Beendigung des Programms wird der Anwender angefragt, ob er die aktuelle Konfiguration speichern will oder nicht.

Optionen / Anzeige

Dieser Befehl dient zur Anzeige oder zum Verstecken von verschiedenen Tastenleisten des Programms (Toolbar).

Optionen / Anzeige / Haupt-Tastenleiste

Mit diesem Befehl kann die Haupt-Tastenleiste entweder angezeigt oder versteckt werden.

Optionen / Anzeige / Zusätzliche Tastenleiste

Mit diesem Befehl kann die zusätzliche Tastenleiste entweder angezeigt oder versteckt werden.

Optionen / Anzeige /Bauteil-Optionen vor der Aktion auf dem Bauteil

Mit diesem Befehl kann die Anzeige des Fensters **Bauteil-Optionen vor der Aktion auf dem Bauteil,** d.h. bei der Wahl des Aktionsmenüs auf dem Bauteil, z. B. Programmieren, erlaubt/verboten werden.

Optionen / Fehler anzeigen

Diese Option ermöglicht, die Form der Fehler-Anzeige als das Resultat des Vergleichs der programmierten Daten einzustellen. Die Fehler-Anzeige kann entweder ausgeschaltet werden, die Fehler können auf dem **Bildschirm** angezeigt werden (max. die ersten 45 Differenzen), oder alle Differenzen können in die **VERIFY.ERR**-Datei auf der Festplatte im aktuellen Inhaltverzeichnis gespeichert werden. Ist die Fehler-Anzeige **ausgeschaltet**, wird die Fehlermeldung vom Steuerprogramm nur im INFO-Fenster angezeigt.

Diese Einstellung wird auf die Festplatte mit dem Befehl **Optionen / Optionen Speichern** gespeichert. Die Default-Einstellung ermöglicht, die Fehler auf dem Bildschirm anzuzeigen.

Optionen / Programmer suchen

Dieser Befehl erlaubt, während des Programmablaufs einen neuen Programmer-Typ sowie die Kommunikationsparameter einzustellen. Zur Verfügung stehen folgende Optionen:

Programmer – es wird ein neuer Programmer-Typ eingestellt, der gesucht wird. Ist **Alle suchen** eingestellt, bedeutet es, dass das Steuerprogramm nach allen unterstützten Programmern suchen wird.

Kommunikation herstellen – ermöglich **manuelle** oder **automatische** Einstellung der Kommunikations-geschwindigkeit für den gewählten Programmer.

Geschwindigkeit – hier wird die Geschwindigkeit eingestellt, falls die manuelle Einstellung der Kommunikations-geschwindigkeit

erlaubt wird, mit der die Daten vom PC in den Programmer gesendet werden. Die Geschwindigkeit ist als Prozentzahl der Maximalgeschwindigkeit ausgedrückt.

Die Änderung der Kommunikationsgeschwindigkeit ist für die PC mit „langsamen" LPT-Ports bestimmt (Laptop, Notebook, usw.). Benutzen Sie diese Möglichkeit in dem Falle, dass sich der Programmer nicht meldet, obwohl er am PC angeschlossen ist, oder wenn die Kommunikation mit dem Programmer nicht zuverlässig ist.

Wird die Kommunikationsgeschwindigkeit automatisch eingestellt, bedeutet es, dass von dem Steuerprogramm die Maximalgeschwindigkeit eingestellt wird.

Port – erlaubt einen Parallelport auszuwählen, in dem der neue Programmer gesucht wird. Sind **Alle Ports** gewählt, bedeutet es, dass der Programmer vom Steuerprogramm in allen Parallelports gesucht wird, die an den Standardadressen zugänglich sind.

Spezielle Portadresse – stellt die Parallelport-Adresse in dem Falle dar, dass ein **spezieller Port** ausgewählt wurde**.**

Mit dem Drücken der **<Enter>** Taste oder der **OK**-Taste wird das Suchen des gewünschten Programmers in Abhängigkeit von den voreingestellten Parametern aktiviert. Es läuft dasselbe Prozess ab, wie beim Starten des Steuerprogramms. Die Liste der voreingestellten Bauteile wird gelöscht und es bleibt nur das aktuelle Bauteil, falls dieses vom neuen Programmer unterstützt wird.

Diese Einstellung wird auf die Festplatte mit dem Befehl **Optionen /Optionen speichern** gespeichert.

Optionen / Handler

Im Dialog **Handler** kann der Handler-Typ und die Kommunikation des Handlers mit dem Steuerprogramm angewählt werden. Der Handler ist eine externe Einrichtung zur Steuerung von Aktien auf Bauteilen im Steuerprogramm. Wenn die Wahl des Handler-Typs auf **Nicht vorhanden** gestellt ist, so befindet sich das Programm im Standardmodus der Steuerung von Aktien auf Bauteilen. Das heißt, der Anwender steuert die Aktien auf Bauteilen direkt mit der Tastatur oder mit der Maus. Falls ein konkreter Handler gewählt wurde, befindet sich das Programm im Spezialmodus, und die Aktien auf Bauteilen sind in Kooperation mit dem Handler automatisch gesteuert.

Der Handler-Dialog enthält folgende Bestandteile:

Gewählter Handler – Einstellung des gewünschten Handlers
Auf dem Port suchen – ermöglicht den Serien-Port zu suchen, wo der gewünschte Handler gesucht wird.

Mit dem Drücken der **<Enter>** Taste oder der **OK**-Taste wird das Suchen des gewünschten Handlers in Abhängigkeit von den voreingestellten Parametern aktiviert. Diese Einstellung wird auf die Festplatte mit dem Befehl **Optionen /Optionen speichern** gespeichert.

Optionen / Modul-Optionen

Dieser Befehl wird bei Mehrfach-Programmern zur Einstellung des MASTER-Sockels und der Aktivität einzelner Sockel verwendet. Im Teil **MASTER** wird der Sockel eingestellt, der zum Lesen vom Bauteil verwendet wird. Im Teil **Sockel erlauben / verbieten** wird die Erlaubnis oder Verbot einzelner Sockel eingestellt. Verbotene Sockel werden bei allen Tätigkeiten auf den Bauteilen ignoriert.

Optionen / Automatisches JA!

Dieser Befehl wird zur Einstellung des **Automatisches JA!** benutzt. In diesem Modus reicht für die Wiederholung der zuletzt durchgeführten Operation nur das neue Bauteil in den ZIF-Sockel einzusetzen. Programm erkennt automatisch das Einsetzen eines neuen Bauteils und führt die letzte Operation durch, ohne dass eine Taste gedrückt werden muss. Das Einsetzen des Bauteiles in den ZIF-Sockel wird auf dem Bildschirm angezeigt. Die wiederholte Ausführung der Operation kann durch die Betätigung der **<Esc>**-Taste während des Wartens auf das Einsetzen/Entfernen des Bauteils in den/aus dem ZIF-Sockel erfolgen.

Anmerkung: *Während des Wartens auf das Einsetzen des Bauteils in den ZIF-Sockel blinkt auf dem Programmer die LED „BUSY"-Kontrollleuchte .*

Dieser Modus kann mit **Automatisches JA!** freigegeben oder ausgeschaltet werden. Wird ein neuer Programmer ausgewählt, wird dieser Modus mit **Optionen / Programmer suchen** ausgeschaltet.

In der **Reaktionszeit** kann der Zeitabstand, während dessen das Bauteil im ZIF-Sockel liegen muss, damit das Einsetzen eines neuen Bauteils vom Programm akzeptiert wird, verlängert werden. Es wird ein Standardzeitabstand voreingestellt. Wird ein Sockel-Adapter benutzt, wird es empfohlen, einen **verlängerten** Zeitabstand zu verwenden.

In **Pins mit Kondensatoren** kann die Liste der Pins eingegeben werden, die mit einem Kondensator, bzw. mit mehreren Kondensatoren gekoppelt sind (z.B. wenn ein Konverter eingesetzt wird, dessen Sperrkondensator zwischen VCC und GND angeschlossen ist), von denen die Probleme bei der Detektion des neu eingesetzten Bauteils in den ZIF-Sockel hervorgerufen werden konnten.
Die Liste der Bauteils-Pins wird in folgender Form angegeben:

pinA, pinB, pinC...

Beispiel: 4,6,17

Durch Auswahl eines neuen Bauteils mit dem Befehl **Bauteil/ Auswahl** oder **Bauteil/Auswahl aus vorgegebenen Bauteilen** wird diese Liste gelöscht.

Diese Einstellung wird auf die Festplatte mit dem Befehl **Optionen / Optionen speichern** gespeichert.

Optionen / Log-Datei

Diese Option hängt sehr eng mit der Benutzung des **Log-Fensters** (Log window) zusammen. Alle Meldungen, die in dieses Fenster eingetragen werden, können auch in die **LOG-Datei** eingetragen werden. Die LOG-Datei besitzt einen gleichbleibenden Namen - **Report.rep** – und sie wird in dem aktuellen Inhaltverzeichnis gebildet.
Die Einstellung **Neu** ruft das Löschen der alten **Report.rep-Datei** in dem Falle auf, dass diese Datei vorhanden ist, und zugleich wird eine neue Datei gebildet.
Die Einstellung **Anhängen** addiert alle Meldungen in die vorhandenen Dateien. Ist die Datei nicht vorhanden, wird sie erzeugt.
Die Einstellungen werden nur beim Programmstart angewandt.
Diese Einstellung kann auf die Festplatte mit dem Befehl **Optionen / Optionen Speichern** gespeichert werden. Die Default-Einstellung schaltet die Benutzung der LOG-Datei aus. Alle Meldungen werden nur im LOG-Fenster angezeigt.

Optionen / Geschützter Modus

Geschützter Modus ist ein Spezialmodus des Programms, wenn im Programm die Tätigkeiten geblockt sind, mit denen der Pufferinhalt oder Einstellungen des aktuell gewählten Bauteils verändert werden können. Der geschützte Modus dient dazu zu verhindern, dass der Anwender irrtümlich den Puffer oder die Bauteil-Einstellung modifiziert, vor allem beim Programmieren einer höheren Anzahl gleicher Bauteile hintereinander, was die Aufmerksamkeit des Anwenders schwächen kann.

Das Programm kann auf folgende Art in den geschützten Modus umgeschaltet werden:

1. Mit Hilfe des Menüs **Optionen / Geschützter Modus**, wo ein Dialog dargestellt wird, der den Anwender auffordert, ein Password einzugeben, das zum Schutz vor unerwünschtem Umschalten aus dem geschützten in den Standardmodus dient. Das Password muss bei Umschalten in den geschützten Modus zweimal bestätigt werden.

2. Beim Einlesen des Projektes, das im geschützten Modus gespeichert wurde.

Das Programm kann aus dem geschützten in den Standardmodus mit Hilfe des Menüs **Optionen / Standardmodus** umgeschaltet werden, wobei der Anwender aufgefordert wird, ein Password einzugeben. Nach dem Eingeben und der Bestätigung des

richtigen Passwords wird der geschützte Modus des Programms aufgehoben, und das Programm arbeitet im Standardmodus. Eine weitere Möglichkeit, wie der geschützte Modus des Programms aufgehoben werden kann, ist die Beendigung des Programms. Das Programm befindet sich im geschützten Modus nur bis zu seiner Beendigung. Beim erneuten Start befindet sich das Programm im Standardmodus (das gilt jedoch nicht, wenn in der Befehlszeile beim Programmstart ein geschütztes Projekt eingegeben wurde, d.h. ein Projekt, das im geschützten Modus gespeichert wurde, und bei dessen Einlesen das Programm in den geschützten Modus umschaltet).

Optionen / Optionen Speichern

Dieser Befehl speichert alle Einstellungen des Programms, deren Speichern aktuell unterstützt wird. Die Einstellungen sind gespeichert, auch wenn das automatische Speichern der Einstellungen bei der Programmbeendigung ausgeschaltet ist. Es werden folgende Optionen gespeichert: Einstellungen im Menü **Optionen**, zehn zuletzt ausgewählte Bauteile, Position und Größe des Hauptprogramm-Fensters, Dateigeschichte.

Diagnostik

Dieser Menü-Befehl enthält den Selbsttest für die Programmer sowie den IO-Test.

Diagnostik / Selbsttest

Mit diesem Befehl wird der Selbsttest des aktuell gewählten Programmers ohne Einsatz des Diagnostikkopfes durchgeführt. Um die Zuverlässigkeit des Selbsttestes zu erhöhen, empfehlen wir ebenfalls **Diagnostik / Selbsttest plus** des Programmers durchzuführen.

Diagnostik / Selbsttest plus

Mit diesem Befehl wird der Selbsttest des aktuell gewählten Programmers mit Einsatz des Diagnostikkopfes durchgeführt. Der Diagnostik-Kopf wird mit dem Programmer geliefert. Wir empfehlen, diesen Test so oft wie möglich, z.B. 1x monatlich, durchzuführen.

Diagnostik / IO test

Dieser Befehl aktiviert einen Testabschnitt für das Testen von integrierten Schaltungen, die in der Abhängigkeit von der Kompatibilität in einige Bibliotheken verteilt sind, die sich in der Lieferung befinden. Der Befehl ermöglicht, die entsprechende Bibliothek, das gewünschte Bauteil sowie die Betriebsart für den Test-Vektoren-Ablauf zu wählen (**Zyklisch oder Einzelschritt**). Steuerungssequenz sowie Testresultate werden im LOG-Fenster (LOG WINDOW) angezeigt.

Diagnostik / Diagnostik-Protokoll anfertigen

Der Befehl **Diagnostik-Protokoll anfertigen** wird zum Generieren diagnostischer Informationen (Einstellung des Bauteils, des Programmers, usw.) in das LOG-Fenster und zum Kopieren des aktuellen Inhalts des ganzen LOG-Fensters in den Speicher verwendet. Aus dem Speicher kann der Inhalt des LOG-Fensters in einen beliebigen Textverarbeitungsprogramm übertragen werden. Das Diagnostik-Protokoll ist besonders dann günstig, wenn es im Steuerprogramm oder im Programmer zu einem Fehler kam, der in Zusammenarbeit mit dem Hersteller behoben werden muss. In einem solchen Fall, wenn man der Kunde eine Fehlermeldung an den Hersteller schickt, ist es empfehlenswert, auch das Diagnostik-Protokoll mitzuschicken, was dem Hersteller ermöglicht, die Ursache schneller zu lokalisieren und den Fehler anschließend zu beheben.

Hilfe

Durch Betätigung der **<F1>**-Taste kann Hilfe aufgerufen werden. Falls Sie ein Menü-Punkt aktivieren und dann **<F1>**- Taste betätigen, wird eine kontext-empfindliche Hilfe aufgerufen.
Nach Betätigung der **<F1>**-Taste während einer Operation mit dem Programmer wird keine Aktion geleistet.

Folgende Hilfe-Menü-Punkte sind markiert:
- Wörter, die die Tasten bezeichnen
- andere wesentliche Wörter
- aktuelle Bezugnahme: klicken Sie auf diese Bezugnahme, um weitere Informationen zu bekommen.

Detaillierte Informationen über Menü-Befehle sind in der integrierten on-line Hilfe zu finden.

Anmerkung:
Eventuelle Abweichungen von dieser Bedienungsanleitung und Verbesserungen können Sie unter www.elnec.com finden.

Hilfe / Unterstützte Bauteile

Dieser Befehl bildet das Verzeichnis aller Bauteile ab, die mindestens durch einen der unterstützten Programmer unterstützt werden. Das „g_"-Präfix vor der Bezeichnung des Bauteils bedeutet, dass das betreffende Bauteil durch einen multifachen Programmer unterstützt wird.

Hilfe / Unterstützte Programmer

Mit diesem Befehl bekommen Sie Informationen über Programmer, die durch das Steuerprogramm unterstützt werden.

Hilfe / Verzeichnis unterstützter Bauteile (aktueller Programmer)

Dieser Befehl generiert das Verzeichnis von Bauteilen, die durch den aktuellen Programmer unterstützt werden, und legt es in Textform in die ?????DEV.TEX-Datei im Verzeichnis, wo sich das Steuerprogramm befindet.

Hilfe / Verzeichnis unterstützter Bauteile (Bezugnahme)

Dieser Befehl generiert das Verzeichnis aller Bauteile, die durch die auf dem Markt verfügbaren Programmer und durch das aktuelle Steuerprogramm für Programmer (weiter nur als Steuerprogramm angeführt) unterstützt werden. Das Verzeichnis im HTML-Format besteht aus folgenden Dateien:

- eine Hauptdatei, **TOP_DEV.htm**, enthält Hersteller unterstützter Bauteile
- weitere HTML-Dateien enthalten Verzeichnisse unterstützter Bauteile einzelner Hersteller.

Die HTML-Hauptdatei befindet sich im Verzeichnis, in dem sich dieses Steuerprogramm befindet.

Weitere HTML-Dateien sind im Unterverzeichnis **DEV_HTML** gespeichert, das sich im Verzeichnis mit dem Steuerprogramm befindet.

Hilfe / Über das Programm

Mit diesem Befehl wird das Fenster mit Grundinformationen über die Programmversion, das Copyright und die Adresse des Lieferanten generiert.

Gemeinsame Hinweise

Software

PG4U/PG4UW ist das gemeinsame Steuerprogramm für alle ELNEC-Programmer. Aus diesem Grunde ist es möglich, dass während der Arbeit mit dem Programm Menüpunkte gefunden werden, die mit dem vorhandenen Programmer nichts zu tun haben.

Das Programm ermöglicht es, die gewählte Datei dem jeweiligen Bauteil zuzuordnen. Diese Datei kann automatisch in den Puffer eingelesen werden, falls der Menüpunkt **Bauteil/BauteilOptionen/ZugeordneteDatei/Automatisches Einlesen** auf „Ja“ eingestellt ist. Als zugeordnete Datei kann die aktuell im Puffer eingelesene Datei eingestellt werden, oder sie kann im Pfad unter dem Menüpunkt **Bauteil/ BauteilOptionen/Zugeordnete Datei/Datei** festgelegt werden.

Automatisches JA! - Eigenschaft des Programms, die es ermöglicht, die letzte Operationen mit dem Bauteil zu wiederholen, ohne das die **JA!**-Taste des Programmers gedrückt werden musste. Das Programm ist selbst in der Lage, das Einlegen eines neuen Bauteils zu erkennen und die letzten Operationen auszulösen. Einstellungen des automatischen JA! sind unter dem Menüpunkt **Einstellungen** zu finden. In dem Fall, dass Sie den Packungskonverter benutzen, bei dem einige PINs mit Sperrkondensatoren belegt werden, tragen Sie bitte diese PINs in die jeweilige Zeile ein. Das automatische JA! steht nur den JetProg-, LabProg+- und SmartProg-Programmern zur Verfügung.

Einige spezielle Bauteile (z.B. Coolrunner-Gruppe der Firma Philips) erfordern eine externe DAT-Datei, die nicht zu Standard-Software-Dateien gehört, die auf einer CD-ROM geliefert werden. In dem Falle, dass Sie solche Bauteile programmieren müssen, stehen Ihnen diese Dateien auf der Internet-Seite: www.elnec.com, Teil Download, zur Verfügung.

Hardware

Weil es eine große Menge von Parallelportarten gibt, könnte es passieren, dass unser Programmer nicht in der Lage ist, mit Ihrem PC zu kommunizieren. Sind Sie ein Anwender, der ein Problem mit der Kommunikation zwischen PC und dem Programmer hat, bzw. eine unzuverlässige Kommunikation zwischen dem PC und dem Programmer hat, überprüfen Sie bitte, ob das Problem der Kommunikation ebenfalls bei anderen Computern, bzw. an anderen Parallelports auftritt.
Ist das nicht der Fall, notieren Sie bitte die genaue Konfiguration Ihres Computers (Typ, Hersteller, Geschwindigkeit, Betriebssystem, benutzte Programme; Hersteller, bzw. der IO-Typ Ihres Parallelportes) sowie andere Umstände, die mit diesem Problem zusammenhängen könnten, und kontaktieren Sie den Hersteller. Benutzen Sie bitte dabei das **DEVICE PROBLEM REPORT**-Formular**,** das der Anlage A beigefügt ist.

ISP (In-System Programming)

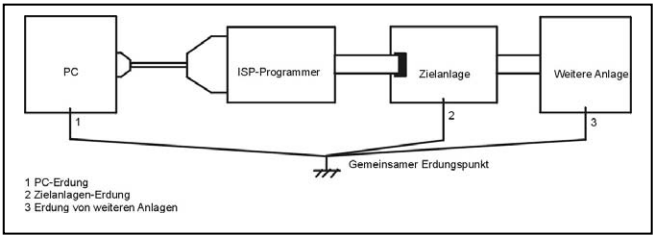
Definitionen

In-system programming ermöglicht das Programmieren und das Umprogrammieren des Bauteils direkt im Gerät ohne es herausnehmen zu müssen. Durch Verwendung einer einfachen Schnittstelle kommuniziert der ISP-Programmer serienmäßig mit dem Bauteil, ist auch imstande, das Gedächtnis auf dem Chip umzuprogrammieren. ISP eliminiert physische Herausnahme des Bauteils aus dem Gerät, was Geld, Zeit,... spart. Es ist für Entwicklungsarbeiten und für Software-Update oder Parameter der Anlagen in der Praxis geeignet.
Das Zielbauteil ist das Bauteil (Mikroprozessor, PLD...), das In-System programmierbar ist.
Die Zielanlage ist die physische Leiterplatte, die das Bauteil enthält, welches In-System programmierbar ist.
Der ISP-Programmer ist ein Programmer, der fähig ist, ISP-Programmierung vorzunehmen (z.B. SmartProg, T51prog, PIKprog+...)

Allgemeine Regeln für In-System programming

Wir empfehlen, folgende Regeln zu beachten, um eine eventuelle Beschädigung des PC, ISP-Programmers oder des Zielsystems zu verhindern.

- Sichern Sie den gemeinsamen Erdungspunkt für die Zielanlage, den ISP-Programmer und den PC.
- Für einen Laptop oder andere PC-Typen, die nicht an den gemeinsamen Erdungspunkt angeschlossen sind, machen Sie mit einem ausreichend dicken Leiter einen Anschluss an einen gemeinsamen Erdungspunkt (verwenden Sie z.B. LPT oder COM-Steckverbinder).
- Alle mit der Zielanlage verbundenen Anlagen müssen auch an den gemeinsamen Erdungspunkt angeschlossen werden.



Vorgehen beim Anschluss des ELNEC ISP-Programmers an die Zielanlage

Während des In-System-Programmierens sind zwei elektrische Anlagen verbunden – der ISP-Programmer und die Zielanlage. Ein nicht qualifizierter Anschluss kann diese Anlagen beschädigen, oder gar vernichten.
Anmerkung: *Sollten Sie die folgenden Anweisungen nicht einhalten und den ISP-Programmer während des In-System Programmierens beschädigen, wird diese Beschädigung als eine durch unsachgemäße Manipulierung entstandene Beschädigung betrachtet. Solche Fälle sind aus der gewährten Garantie ausgenommen.*

1. Schalten Sie beide Anlagen aus – sowohl den ISP-Programmer, als auch die Zielanlage.
2. Sichern Sie gleiches GND-Potential für alle Anlagen, z.B. schließen Sie GND aller Anlagen zusammen.
3. Schieben Sie einen Steckverbinder des ISP-Kabels in den ISP-Programmer.
4. Im Steuerprogramm wählen Sie das Zielbauteil und die mit dem Bauteil zusammenhängenden Einstellungen.
5. Starten Sie Aktion auf dem Zielbauteil (Lesen, Programmieren...).
6. Gehen Sie nach dem Steuerprogramm vor, und auf seine Anweisung schließen Sie das andere Ende des ISP-Kabels in die Zielanlage und schalten Sie diese ein.
7. Auf die Anweisung des Steuerprogramms koppeln Sie das ISP-Kabel von der Zielanlage ab, und schalten Sie diese aus.
8. Wenn Sie weitere Operationen mit dem Zielbauteil durchführen möchten, setzen Sie mit dem Punkt 5 fort.

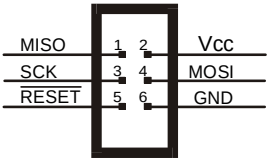
Empfehlungen zum Entwurf der Zielanlage mit dem ISP-programmierbaren Bauteil

Die Zielanlage muss so entworfen werden, damit die zum Programmieren notwendigen Signale zum ISP-Programmer über den ISP-Steckverbinder direkt führen. Falls die Zielanlage die für das In-System Programmieren notwendigen Signale auch zu anderen Zwecken verwendet, ist es notwendig, diese Signale ausreichend zu trennen, d.h. die Zielanlage darf diese Signale während des ISP-Programmierens nicht beeinflussen.
Für die In-System programmierbaren Bauteile veröffentlichen die Hersteller Applikationshinweise. Die Programmer der Firma ELNEC sind so entworfen, damit bei der Einhaltung dieser Applikationshinweise keine Probleme beim Programmieren der Bauteil über den ISP-Steckverbinder auftreten. Die Bedingung ist jedoch die exakte Einhaltung dieser Empfehlungen. Applikationshinweise, die ELNEC beim ISP verwendet, sind auf www.elnec.com , Teil Applikationshinweise, veröffentlicht.

Beispiele von Applikationshinweisen

Atmel AVR- und AT89Sxxx-Prozessoren

Dieser Applikationshinweis ist im T51prog verwendet. Der Anschluss korrespondiert mit dem Applikationshinweis der Firma Atmel AVR910: In-System Programming. Dieser Applikationshinweis beschreibt den Anschluss des ISP-Steckverbinders in der Zielanlage (Blick von oben).



Beschreibung von erwünschten Pins fürs ISP-Programmieren

Pin	Bezeichnung	Kommentar
SCK	Serienuhr	Programmieruhr, generiert durch den ISP-Programmer (Master).
MOSI	Master Out – Slave In	Kommunikationslinie aus dem ISP-Programmer (master) in das Ziel-MCU, das zu programmieren ist (slave).
MISO	Master In – Slave Out	Kommunikationslinie aus dem Ziel-MCU (slave) in den ISP-Programmer (master).
GND	Gemeinsame GND	Anlagen müssen gemeinsame GND haben.
RESET	Reset des Ziel-MCU	Zum Aktivieren des ISP-Programmierens muss im Ziel-MCU der Reset auf aktivem Niveau gehalten werden. Der ISP-Programmer sollte den Reset des Ziel-MCU steuern.
Vcc	Speisung	Wegen Programmieren bei verschiedenen Spannungen kann der ISP-Programmer aus der Zielanlage gespeist werden. Der ISP-Programmer kann während des Programmierens das Ziel-MCU speisen.

PICmicro-Prozessoren

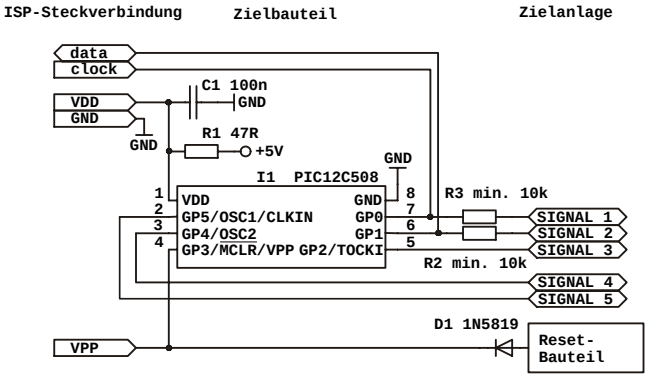
Dieser Applikationshinweis wurde im PIKprog+ verwendet. Der Anschluss korrespondiert mit den Applikationshinweise der Firma Microchip TB013, TB017, TB016: How to Implement ICSP™ Using PIC16CXXX OTP (PIC12C5XX OTP) (PIC16F8X Flash) MCUs. Diese Anmerkungen beschreiben die Anforderungen an die Zielanlage mit dem In-System programmierbaren Bauteil und an den ISP-Programmer.

Am ISP-Programmieren von PICmicro-Prozessoren sind folgende Signale beteiligt:

MCLR\ / VPP	Reset / Überführung in den Programmiermodus
RB6 (GP1)	Uhr
RB7 (GP0)	Dateninput / -output
VDD	Speisespannung
GND	Erdung

Wenn 12 V an den MCLR\ / VPP-Pin geführt, gerät das Bauteil in den Programmiermodus. Dieses Signal muss vom Rest der

Zielanlage getrennt werden, damit es nicht zu seiner Vernichtung (oder der Vernichtung des Programmers) kommt.
RB6- und RB7-Signale beteiligen sich direkt am Programmieren, und ihre Beeinflussung durch die Zielanlage während des Programmierens würde zu Fehlern beim Programmieren führen.
Beim Vergleich von Daten, die in den PICmicro-Prozessor programmiert wurden, wird der Vergleich von Stromversorgungsgrenzwerten verwendet, d.h. Vergleich bei minimal und maximal erlaubter Speisespannung. Deswegen ist es notwendig, auch das VDD pin PICmicro des Prozessors von dem Rest der Zielanlage zu trennen.
Dieser Anschluss wird von der Firma ELNEC empfohlen.



Sonstiges

Hinweise zu Windows 3.11/95/98/Me/NT/2000/XP:
Um die korrekte Funktion des Steuerprogramms und des Programmers zu gewährleisten, muss der Parallelport, an dem der Programmer angeschlossen ist, ausschließlich für diesen Programmmer reserviert werden. Dieser Port darf nicht von anderen im Computer laufenden Programmen benutzt werden (z.B. Scanner-Software usw.).

Von den Steuerprogrammen PG4U/PG4UW werden alle Modi des LPT-Portes unterstützt und aus diesem Grunde ist es nicht notwendig, den LPT-Port auf die richtige Kommunikation mit dem Programmer zu konfigurieren.

WIN98 enthält ein Problem; und zwar in der Datei MSDOS.SYS. Die Anfangseinstellung der Variable „DoubleBuffer“ ist 1, und aus diesem Grunde laufen die DOS-Applikationen ziemlich langsamer. Ändern Sie bitte diesen Wert auf DoubleBuffer = 0 ab.

Falls Sie das Windows-Programm benutzen, verschieben Sie bitte kein Fenster, wenn die LED „BUSY“ leucht. Es könnte passieren, dass die Kontrollschaltungen der Kommunikation ansprechen und die Fehlermeldung **Kommunikationsfehler** erscheint.

Wenn Probleme auftreten Gewährleistungs-bedingungen

Falls im Zusammenhang mit dem Programmer Probleme auftreten

Wir bemühen uns auf jeden Fall, dass unsere Kunden mit unseren Erzeugnissen zufrieden sind. Trotzdem kann es aber passieren, dass Probleme auftreten, die wir nicht vorgesehen haben, bzw. die wir einfach nicht beeinflussen können. Folgen Sie bitte den unten angeführten Anweisungen, bevor Sie wegen der Installation des Programmers Kontakt mit Ihrem Händler oder direkt mit uns aufnehmen, bzw. bevor Sie das Erzeugnis, das Ihrer Meinung nach fehlerhaft ist, zur Reparatur schicken.

- Versuchen Sie bitte, die Antwort auf Ihre mit dem Programmer zusammenhängenden Probleme auf die einfachste Art zu finden.
  - Lesen Sie bitte sorgfältig die Dokumentation durch, die dem Programmer entweder in Schriftform, oder in Form der elektronischen Medien beigefügt ist. Vielleicht finden Sie in dieser Dokumentation die Antwort, wie das entstandene Problem zu lösen ist.
  - Versuchen Sie bitte, den Programmer an einen anderen Computer anzuschließen. Ist die Installation erfolgreich, vergleichen Sie bitte beide Computer miteinander, und aufgrund des Resultats versuchen Sie bitte, das Problem zu beheben.
  - Sie können ebenfalls den Fachmann im Bereich Computertechnik, der Ihren Computer betreut, ansprechen.
  - Konsultieren Sie bitte Ihr Problem mit denen, die einen Programmer bereits installiert haben, bzw. die denselben Programmer verwenden.
- Hilft Ihnen die beigefügte Dokumentation, bzw. die unter den restlichen Punkten beschriebenen Anweisungen bei Lösung Ihres Problems nicht, kontaktieren Sie bitte direkt die Firma ELNEC. Viele Probleme können telefonisch, per Fax oder per E-Mail geklärt werden, falls Sie mit uns Kontakt aufnehmen wollen:
  - **per Post/per Fax** – Kopieren Sie bitte das **DEVICE PROBLEM REPORT**-Formular und füllen Sie dieses sorgfältig nach Anweisungen aus, die im unteren Teil dieses Formulars angeführt sind. Geben Sie bitte alle Unstimmigkeiten an, auch scheinbar nicht wichtiges, fehlerbedingtes Verhalten des Programmers und/oder des Steuerprogramms. Wir bitten Sie, das ausgefüllte Formular per Fax an die Faxnummer zu senden, die sich im Steuerprogramm PG4U, Sektion **Info, bzw.** PG4UW, Sektion **Hilfe / Über das Programm** befinden**.**
  - **Per E-Mail** – Kopieren Sie bitte das **DEVICE PROBLEM REPORT**-Formular, das sich auf der beigefügten CD befindet und füllen Sie dieses in einem beliebigen Editor nach Anweisungen, die im unteren Teil des Formulars angeführt sind, sorgfältig aus. Geben Sie bitte alle Unstimmigkeiten ein, auch scheinbar nicht wichtiges, fehlerbedingtes Verhalten

des Programmers und/oder des Steuerprogramms. Wir bitten Sie, das ausgefüllte Formular per E-Mail auf die E-Mail-Adresse: **elnec@elnec.com** zu übersenden.

- **Telefonisch** - Kopieren Sie bitte das **DEVICE PROBLEM REPORT**-Formular und füllen Sie dieses sorgfältig nach Anweisungen aus, die im unteren Teil dieses Formulars angeführt sind. Geben Sie bitte alle Unstimmigkeiten ein, auch scheinbar nicht wichtiges, fehlerbedingtes Verhalten des Programmers und/oder des Steuerprogramms. Wir bitten Sie, das ausgefüllte Formular per Fax an die in der Sektion „**Info“** des PG4U-Steuerprogramms, in der Sektion **Help/About** des PG4UW-Steuerprogramms angeführte Faxnummer zu übersenden. Erst dann rufen Sie bitte den Kundenservice der Firma ELNEC an. Verwenden Sie bitte immer die Telefonnummer, die Sie auf derselben Stelle im Steuerprogramm finden. Bewahren Sie bitte die Betriebsanleitung, den Programmer und auch das ausgefüllte **DEVICE PROBLEM REPORT**-Formular in ihre Nähe auf, damit Sie auf die Fragen der Kontaktperson schnell reagieren können. Sollten wir nicht in der Lage sein, den Fehler auf diese Art zu beheben, werden wir nach einer anderen und für Sie akzeptablen Art der Problemlösung suchen.

- Sollte festgestellt werden, dass der Programmer repariert werden muss, schicken Sie bitte in dem Falle den Programmer zur Reparatur an den Kundenservice, Auslandsvertretung der Firma ELNEC, bzw. direkt an die Firma ELNEC. Die Sendung sollte enthalten:
  - fehlerhaftes Produkt mit komplettem Zubehör
  - ausgefülltes **DEVICE PROBLEM REPORT**-Formular und
  - Kopie der Rechnung mit Verkaufsdatum.

Ohne dass diese Voraussetzungen erfüllt werden, kann das Produkt nicht zur Reparatur angenommen werden!

Anmerkungen:

- *Das **DEVICE PROBLEM REPORT**-Formular können Sie:*
- *in der **Anlage „A“** dieser Betriebsanleitung und*
- *auf der CD (Datei **DEVPRORS.TXT**) finden.*
- *In dem Falle, dass die für ein Halbjahr gewährte Gewährleistung bereits abgelaufen ist, ist dem reklamierten Produkt ebenfalls der Beleg "**Erweiterte Gewährleistung“** mit angegebener Registernummer als Nachweis für die Erweiterung der Gewährleistung seitens der Firma ELNEC beizufügen. Nähere Informationen entnehmen Sie bitte dem Kapitel "**Gewährleistungsbedingungen**".*

Was muss man tun, wenn festgestellt wird, dass die Bauteile nicht unterstützt werden?

In der Praxis kann es passieren, dass einige Bauteile von Ihrer Version des Steuerprogramms zum Programmer momentan nicht unterstützt werden. Außerdem kann es passieren, dass während der Arbeit mit Bauteilen, die vom Steuerprogramm unterstützt werden, Probleme auftreten, wobei der Fehler nicht im Programmer liegt. In solchen Fällen verfahren Sie bitte wie folgt:

- Besorgen Sie sich bitte die neueste Version des Steuerprogramms (am schnellsten per Internet). Vergewissern Sie sich, ob die Unterstützung des Bauteils in diese Version des Steuerprogramms bereits implementiert wurde.

Ist das nicht der Fall, verfahren Sie bitte wie folgt:

- Füllen Sie das **DEVICE PROBLEM REPORT**-Formular nach den Anweisungen aus, die im unteren Teil des Formulars angeführt sind. Geben Sie alle, wenn auch scheinbar unwichtige Details des Falles ein. Das ausgefüllte Formular sowie die Musterbauteile (falls diese vorhanden sind) senden Sie bitte an uns. Nach dem Implementieren der Unterstützung des Bauteils in das Steuerprogramm senden wir Ihnen die Musterbauteile natürlich zurück.

Lesen Sie bitte auch die Beschreibung der „AlgOR“- Dienstleistung in der Anlage dieser Betriebsanleitung.

Hinweis:
*Das **DEVICE PROBLEM REPORT**-Formular ist zu finden:*

- *in der Anlage zu dieser Betriebsanleitung,*
- *auf der beigefügten CD (Datei DEVPRORS.TXT).*
- *an der Internet-Adresse: www.elnec.com, Teil: Applikationshinweise.*

Gewährleistungsbedingungen

Firma ELNEC s.r.o. Prešov, Slowakei gewährleistet die einwandfreie Funktion der Programmer. Sie gewährleistet, dass die Programmer sowie deren Bestandteile weder Materialfehler noch Bearbeitungsfehler ausweisen. Garantiezeit beträgt **drei Jahre** (für JetProg-, BeeProg-, LabProg+-, SmartProg-, PREPROM-02aLV-Programmer), oder **ein Jahr** ab Einkaufsdatum (für MEMprog-, T51prog-, 51&AVRprog-, PIKprog+-, PIKprog- und SEEprog-Programmer). Diese Garantiezeit ist zu 25.000 Zyklen je ein DIL ZIF-Sockel oder zu 10.000 Zyklen je ein anderer ZIF-Sockel begrenzt. Sollte ein Programmer von uns als mangelhaft anerkannt werden, wird er von der Firma ELNEC oder von der autorisierten Kundendienststelle kostenlos repariert, bzw. es werden die fehlerhaften Teile ausgetauscht. Bei bereinigten Fehlern gilt für die ausgetauschten Teile oder für den ganzen Programmer nur die noch nicht abgelaufene Garantiezeit.

Für die Reparatur im Rahmen der Gewährleistung muss das Einkaufsdatum durch entsprechende Rechnung oder Lieferschein nachgewiesen werden. Erweiterte Gewährleistung bezieht sich nur auf die Inhaber von Programmern (Endkunden), die über eine Bescheinigung über die Erteilung der erweiterten Gewährleistung verfügen (dazu siehe Hinweis am Ende des Kapitels "Gewährleistungsbedingungen").Für andere Kunden beträgt die Garantiezeit 6 Monate.

Die Gewährleistung bezieht sich nicht auf übliche Abnutzung oder auf irgendwelche mechanische Beschädigung des Programmers. Aus der Gewährleistung sind ebenfalls Produkte ausgenommen, die nicht bestimmungsgemäß benutzt wurden, oder die durch nicht fachgerechte Handhabung, falsche Installation, oder durch höhere Gewalt beschädigt wurden. Die Gewährleistung kann sich ebenfalls nicht auf die Produkte beziehen, in die ein Eingriff von einer nicht autorisierten Person ausgeführt wurde.

Die Höhe der Reparaturkosten nach Ablauf der Gewähr-leistung ist von dem Preis für reparierte/ausgetauschte Teile, von der Dauer der Reparatur sowie von Höhe der Transportkosten abhängig. Die Firma ELNEC oder der Vertriebhändler sind berechtigt zu entscheiden, ob das Produkt repariert oder ausgetauscht wird, sowie ob die Reparatur im Rahmen der Gewährleistung oder erst nach dem Ablauf der Gewährleistung geleistet wird.

Hinweis:

- *Die Bescheinigung über "**Erweiterte Gewährleistung**" erhält jeder Kunde, der sich bei der Firma ELNEC aufgrund der vollständig ausgefüllten **Registrierkarte** registriert. Diese Registrierkarte sollte per Post, per Fax oder per E-Mail spätestens innerhalb von 4 Wochen ab dem Kaufdatum des Programmers übersendet werden. Von uns bekommen Sie umgehend eine Bescheinigung über die "**erweiterte Gewährleistung**" mit Ihrer Registriernummer. Das Risiko für*

Nichteinreichung der ausgefüllten Registrierkarte trägt der Kunde. Die Registrierkarte ist der Bestandteil der Lieferung des Programmers. Fehlt die Registrierkarte aus irgendeinem Grunde, entnehmen Sie bitte die Kopie aus der Anlage dieser Betriebsanleitung.

- *Lesen Sie ebenfalls die Sektion "Falls im Zusammenhang mit dem Programmer Probleme auftreten".*

Kontaktangaben des Herstellers:

Hersteller:
✉: **ELNEC s. r. o.**, Pošta 5, P. O. Box 22, SK - 08005 Prešov
☎: +42151/77 34 328, 77 31 007, Fax 77 32 797
www.elnec.com, elnec@elnec.com

Anlage

Anlage A - Device Problem Report

Kopieren Sie das A4-Format oder verwenden Sie die Vorlage des A4-Formats, die Bestandteil der Lieferung ist. Das Formular befindet sich auf der Installations-CD und der Internetseite www.elnec.com.

DEVICE PROBLEM REPORT

Betreff (Problembeschreibung):_____ Datum:_____

Kunde
Kunde:_____ Vertriebshändler:_____
Adresse:_____ Einkaufsdatum:_____
Kontaktperson und e-mail:_____ Datum des Einschickens der Reg.karte:_____

Informationen über den Programmer
Programmer (Typ/Ausführung):_____ Speisenetzspannung:____ V (für 51&AVRprog und PIKprog)
Seriennummer: _____ Version des Steuerprogramms PG4U/W:_____
Konfiguration (Module, Konverter): _____
Speisequelle: Geliefert am Eigene Quelle (Outputspannung und -strom): _____

Information über den PC, an dem der Programmer betrieben wird
Hersteller und Ausführung:_____ Desktop Notebook
Prozessor, Geschwindigkeit:_____ LPT-Position: mother board ISA-Karte PCI- Karte
Operationssystem und Version:____ LPT-Port-Karte: Standard ECP/EPP 1284
Grundspeicher/davon frei:_____ LPT-Einstellung: SPP BIDIR EPP ECP

Informationen über das Bauteil, mit dem Sie ein Problem haben
Bauteil-Typ (ganzer Name, inkl. Präfix/Suffix):_____ Packungstyp: Kunststoff keramisch keramisch mit Fenster
Hersteller/Logo:_____ Beschreibung an der ober. _____
Packungen (DIL40, PLCC44, SOIC20, ...):_____ und unter. Seite des Bauteils _____

Dringlichkeit des Problems: bis __ Tagen bis __ Wochen bis __ Monaten
Wie oft arbeiten Sie mit diesen Bauteilen: immer JA/NEIN gelegentlich JA/NEIN
Geschätzte Anzahl von Programmierungen des Bauteils: ca. ___ St. jährlich
Sind Muster zur Verfügung? Ja (ich lege sie bei) Ja Nein

Ergänzende Fragen
- Haben Sie die neueste Version des Steuerprogramms installiert? Ja Nein
- Kennen Sie die Eigenschaften und das richtige Verhalten des Programmers und des Bauteils gründlich? Ja Nein
- Ist der Sockel des Programmers, bzw. des Adapters verunreinigt, eventuell abgelaufen? Ja Nein
- Ist das Problembauteil neu oder gebraucht? Neu Gebraucht
- Tritt der Fehler bei allen Bauteilen dieser Art auf? Ja Nein Ich habe nur ein Bauteil
- Tritt der gleiche Fehler auch bei Bauteilen aus einer anderen Serie auf? Ja Nein Ich habe nur eine Serie
- Bei welcher Tätigkeit mit dem Bauteil tritt der Fehler auf? Programmieren Lesen ID_check
- Ist der Programmer bei anderen Bauteiltypen erfolgreich? Ja Nein
- Tritt der Fehler dauerhaft oder nur zufällig auf? wiederholt zufällig
- Tritt der Fehler auch auf anderen PCs auf? Ja Nein Ich habe es nicht probiert
- Tritt der Fehler mit DOS auch mit der WIN-Version des Programms? Ja Nein
- Wie war das Ergebnis des Selbsttests/Selbsttests+ des Programmers (falls vorhanden)? Fehlerlos Fehlerhaft (Anlage)

Auf dieser Stelle führen Sie bitte die Beschreibung der Tätigkeiten ein, nach denen das Problem erscheint, d.h. führen Sie die einzelnen Schritte, Operationen oder Funktionen in der Reihefolge ein, die die Entstehung des Problems demonstrieren. Beschreiben Sie bitte das Problem ganz ausführlich – ausführliche Beschreibung erhöht die Wahrscheinlichkeit, dass der Fehler in kurzer Zeit behoben wird. Die Problembeschreibung sollte außer anderem auch Auskunft enthalten, was richtig funktioniert, was nicht richtig funktioniert, sowie die Angabe der Fehlermeldungen, die bei einem Fehler des Programms angezeigt werden – legen Sie den Inhalt des ganzen LOG-Fensters bei (bei der WIN-Version der Software kennzeichnen Sie den ganzen Inhalt mit Ctrl+A und kopieren ihn in den Editor, in dem Sie dieses Formular ausfüllen). Wir empfehlen, den Befehl Diagnostik / Diagnostik-Protokoll bilden zu verwenden. Am besten wäre es, wenn Sie uns das problematische Bauteil zur Überprüfung schicken. Geben Sie ebenfalls Ihre Vorstellungen und Erwartungen, bzw. Ihren Vorschlag auf Problemlösung an.

_____
_____
_____
_____
_____

Anmerkung:

- *Falls Sie nicht die neueste Version des Steuerprogramms installiert haben, bekommen Sie diese auf der Internetseite www.elnec.com (Sektion Download).Die neueste Software-Version im Falle von Problemen zu haben ist wichtig, denn:*
- *Es ist möglich, dass das Problem, das Sie haben, mit der Änderung des Programm beseitigt werden kann, und schon gelöst wurde.*
- *Ältere Versionen von Steuerprogrammen haben wir nicht, deswegen kann sich Ihre SW-Version nach einer Empfehlung von uns anders verhalten, und zwar auch im Falle eines guten Programmierers, was die Identifizierung des Problems wirklich erschwert.*
- *Falls Sie Muster aus dem Ausland zuschicken, legen Sie zu Ihrer Sendung auch ein Begleitschreiben mit der Erklärung "kostenloses Muster, Preis zu Zollzwecken 10 EUR" bei.*

Anlage B - Keep-current-Dienstleistung

Der **Keep-Current-Dienstleistung ist eine vorausbezahlte Dienstleistung** mit der Aufgabe, die ELNEC-Programmer sowie die dazugehörige Dokumentation auf dem neuesten Stand (auf die neueste Version) des Steuerprogramms zu bringen.

Im Rahmen dieser Dienstleistung bekommt von uns jeder Kunde die neueste Version des Steuerprogramms sowie die entsprechenden Ergänzungen zurr Bedienungsanweisung (Keep-Current-Paket). Die Keep-Current-Funktion gewährleistet, dass Sie ganz einfach mit geringen Aufwendungen und ohne Probleme die beste Qualität für das Programmieren mit ELNEC-Programmern bekommen werden.

Wir können Ihnen das Keep-Current-Paket wie folgt übersenden :

- per Post
- per E-Mail.

Einzelheiten sind auf www.elcom.com zu finden.

Warum sollte die neueste Version des Steuerprogramms verwendet werden?

- Jede Zeit bringen die Halbleiter-Hersteller neue Bauteile in neuen Packungen, die mit Hilfe der neuesten Technologien hergestellt wurden und die zur Erhöhung der Flexibilität und Produktionsgeschwindigkeit beitragen sollen, auf den Markt. In einem Jahr werden von uns Unterstützungen für ca. 200-500 neue Bauteile in die Steuerprogramme implementiert.
- Während der Lebensdauer eines programmierbaren Bauteils werden mehrere Änderungen des Fertigungsprozesses durchgeführt. Das Ziel dieser Änderungen ist, die Parameter des Bauteils zu halten oder diese noch zu verbessern, und ebenfalls die Effektivität des Fertigungsprozesses zu erhöhen. Diese Änderungen haben sehr oft Änderung des Programmieralgorithmus zur Folge (Programmieralgorithmus bedeutet bestimmte Befehlsequenz, wobei anhand dieser Befehle der Programmer erkennt, wie er die Daten in das jeweilige programmierbare Bauteil einprogrammieren soll). Einsatz der neuesten Algorithmen während des Programmierens gewährleistet das beste Ergebnis dieses Prozesses. Sehr oft passiert es, dass das Programmieren eines Bauteils mit älteren Algorithmen zwar in Ordnung ist, dieser Prozess kann aber nicht die entsprechende Qualität des Programmierungsprozesses gewährleisten. Gute Qualität des Programmierens kann nur in dem Falle sichergestellt werden, dass die Optimalalgorithmen eingesetzt werden. Werden die neuesten Algorithmen nicht eingesetzt, könnte es bedeuten, dass die Effektivität des Programmierens sinkt (d.h. es erhöht sich die Anzahl von fehlerhaft programmierten Bauteilen) und dass sich die Dauer des Programmierens verlängert. Dies kann natürlich auch die

langfristige Datenspeicherung im programmierten Bauteil direkt beeinflussen.

Auf jeden Fall bemühen wir uns, die Unterstützung für neue oder geänderte Bauteile noch vor der Auslieferung oder unmittelbar nach der Auslieferung des Bauteils zu implementieren und auf diese Art sicherzustellen, dass für das jeweilige Bauteil die neuesten Programmieralgorithmen verwendet werden.

Anlage C – AlgOR-Dienstleistung

Algorithmus On Request, Algorithmen auf Wunsch des Kunden

AlgOR-Dienstleistung stellt eine kostenlose Dienstleistung dar, die uns hilft, den Wünschen unserer Kunden flexibel in dem Falle entgegenzukommen, wenn es sich um das Implementieren der Unterstützung für neue programmierbare Bauteile in die vorhandene Version des Steuerprogramms handelt. Diese Dienstleistung steht Ihnen natürlich auch in dem Falle zur Verfügung, wenn Sie Ihr Steuerprogramm um neue Eigenschaften ergänzen wollen.

Der Mechanismus der AlgOR-Dienstleistung ist sehr einfach. Es reicht, wenn Sie uns Ihre Anforderung auf Ergänzung der Unterstützung des X-Bauteils im Steuerprogramm übermitteln (falls möglich, bitten wir auch um Zusendung des Bauteilmusters). Nach dem Implementieren der Unterstützung bekommen Sie von uns die neueste Version des Steuerprogramms mit gewünschten Eigenschaften (und natürlich Ihr Bauteilmuster). Kann die Unterstützung in das Steuerprogramm aus irgendwelchem Grund nicht implementiert werden (zu hohe Aufwendungen, die Programmieralgorithmen sind nicht vorhanden, es müsste ein teuer zusätzliches Modul gebaut werden), werden wir Sie umgehend kontaktieren und für Sie die beste Lösung finden, damit Sie zufrieden sind.

Hinweis:

- *Ihre Anforderung tragen Sie bitte in das "AlgOR-Formular" (Algorithmus On Request) ein. Das ausgefüllte Formular bitten wir Sie direkt an uns zu übersenden.*
- *Die AlgOR-Dienstleistung ist kostenlos, d.h. es bestehen keine Ansprüche der Kunden auf diese Dienstleistung. Wir behalten uns das Recht vor, während der Bearbeitung der neuen Bauteile die Prioritäten der Arbeit nach unserem Ermessen festzulegen.*

Mit diesem Formular kann der Kunde beantragen, ein bis jetzt nicht unterstütztes Bauteil und neue Eigenschaften in das Steuerprogramm zu ergänzen. Füllen Sie das Formular aus und schicken Sie es uns per e-mail, oder per Fax oder per Post direkt an die Firma ELNEC. Ein unvollständig ausgefülltes Formular verstehen wir als eine Information über geringere Wichtigkeit der gewünschten Lösung von seiten des Kunden. Die Unmöglichkeit, die Implementierung der Unterstützung am Muster zu überprüfen hat Verspätung (eventuell auch Unterbrechung) der Bearbeitung Ihrer Anforderung zur Folge.

Kopieren Sie das A4-Format oder verwenden Sie die Vorlage des A4-Formats, die sich in der Lieferung befindet. Das Formular befindet sich auch auf der Installations-CD und im Internet unter www.elnec.com.

AlgOR (Algorithms On Request)

Betreff (Problembeschreibung): _____
Datum: _____

*Kunde:*____ _____
Adresse: _____
_____
Kontaktperson und E-mail: _____
Vertriebshändler: _____
Einkaufsdatum des Programmers: _____
Datum des Abschickens der Registrierkarte: _____

Programmer (Typ/Ausführung): _____
Serien-Nummer: _____
Steuerprogramm/Version: _____

Informationen über das zu programmierende Bauteil
Bauteiltyp (vollständige Bezeichnung): _____
Hersteller/Logo: _____
Packung (DIL40, PLCC44,...): _____

Dringlichkeit des Problems: bis __ Tagen bis __Wochen bis __Monate
Wie oft arbeiten Sie mit diesen Bauteilen: immer J/N gelegentlich J/N
Geschätzte Anzahl von Programmierungen des Bauteils: ca. ___ St. jährlich
Sind Muster zur Verfügung? Ja (ich lege sie bei) Ja Nein

***Spezifizierung der Anforderung**: Beschreibung im Falle der Anforderung auf Veränderung des Steuerprogramms. Führen Sie Ihre Vorstellung an, wie sich das Programm nach der Modifizierung verhalten sollte.*

_____
_____
_____
_____
_____

Anmerkungen

- *Überprüfen Sie bitte vor dem Abschicken des Formulars, ob das gewünschte Bauteil nicht bereits durch die neueste Version des Steuerprogramms unterstützt wird.*
- *Falls Sie Muster aus dem Ausland zuschicken, legen Sie zu Ihrer Sendung auch ein Begleitschreiben mit der Erklärung "kostenloses Muster, Preis zu Zollzwecken 10 EUR" bei.*
- *Dieses Formular kann nur für eine Anforderung für EIN Bauteiltyp, bzw. EINE Modifizierung der Programmausstattung verwendet werden.*

Anlage D - Registrierkarte

Wurde aus irgendeinem Grunde der Sendung keine Registrierkarte beigefügt, übersenden Sie bitte die Kopie der unten angeführten Registrierkarte, bzw. benötigte Informationen direkt an uns. Wir möchten darauf hinweisen, dass ohne gültige Registrierung bei der Firma ELNEC (innerhalb 4 Wochen ab dem Kaufdatum) sich auf die gekauften Programmer nur übliche Gewährleistung bezieht. In dem Falle beträgt die Garantiezeit nur 6 Monate.

✂----------

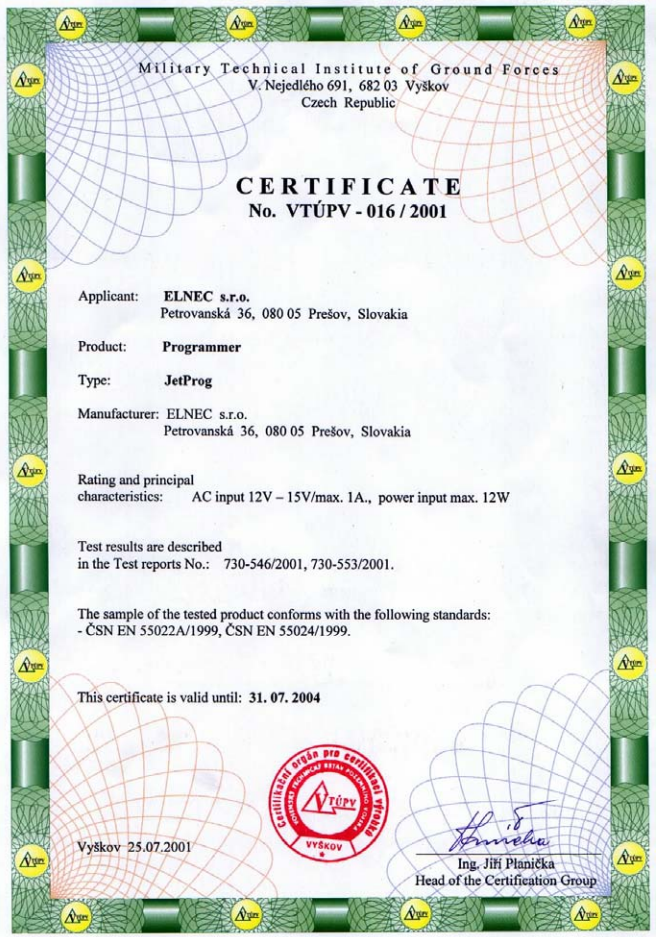
Produkt: _____
Serien-Nummer: _____
Name: _____
Firma: _____
Abteilung: _____
Anschrift: _____
Stadt, PLZ: _____
Land: _____
Telefon/Fax/E-Mail: _____
Einkaufsdatum: _____
Gekauft bei der Firma: _____

Kennzeichnen Sie bitte:

- verwendeter PC-Typ:
  ❒AT/386 ❒486 ❒Pentium I-II ❒ Pentium III-4
- Betriebssystem: ❒MS-DOS
  ❒WINDOWS 3.x ❒WINDOWS 95/98
  ❒WINDOWS NT/2000/XP
- Wie haben Sie über die Firma ELNEC erfahren?
  ❒Anzeige ❒Verkaufsstelle ❒sonstiges _____
- Warum haben Sie sich für dieses Produkt entschlossen?
  ❒Preis ❒Qualität ❒Empfehlung
  ❒eigene Erfahrung _____
- Anmerkungen:_____
  _____
  _____
  _____
  _____

Anlage E – CE Zertifikate

E/1 – CE Zertifikat JetProg



Military Technical Institute of Ground Forces
V. Nejedlého 691, 682 03 Vyškov
Czech Republic

CERTIFICATE
No. VTÚPV - 016 / 2001

Applicant: **ELNEC s.r.o.**
Petrovanská 36, 080 05 Prešov, Slovakia

Product: **Programmer**

Type: **JetProg**

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: AC input 12V – 15V/max. 1A., power input max. 12W

Test results are described in the Test reports No.: 730-546/2001, 730-553/2001.

The sample of the tested product conforms with the following standards:
- ČSN EN 55022A/1999, ČSN EN 55024/1999.

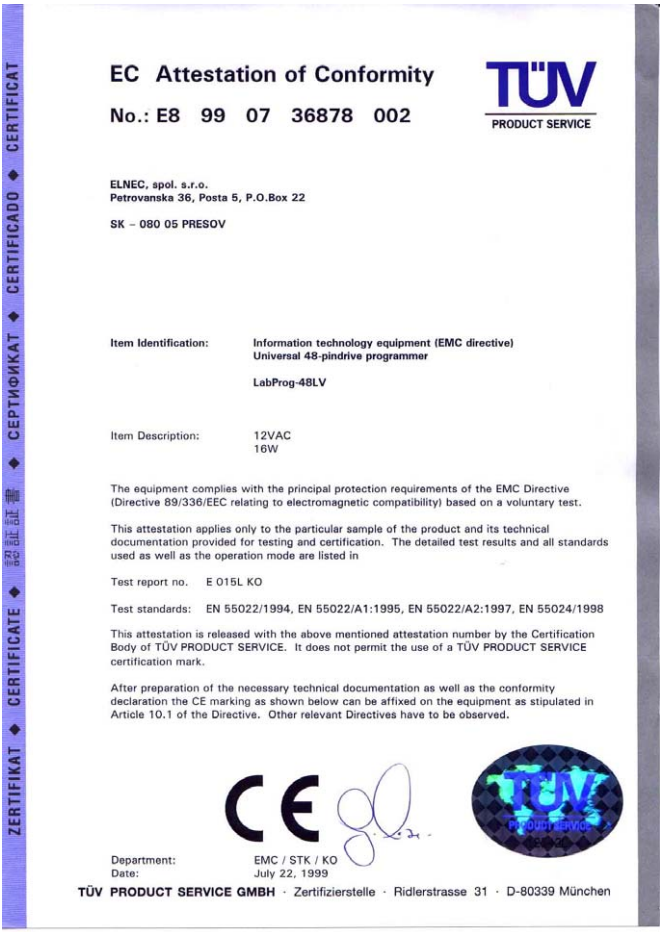
This certificate is valid until: **31. 07. 2004**

Vyškov 25.07.2001

Ing. Jiří Planička
Head of the Certification Group

E/2 – CE Zertifikat LabProg+

LabProg+ ist die neue Bezeichnung von LabProg-48LV.



ZERTIFIKAT ◆ CERTIFICATE ◆ 認証書 ◆ СЕРТИФИКАТ ◆ CERTIFICADO ◆ CERTIFICAT

EC Attestation of Conformity

No.: E8 99 07 36878 002

TÜV PRODUCT SERVICE

ELNEC, spol. s.r.o.
Petrovanska 36, Posta 5, P.O.Box 22

SK – 080 05 PRESOV

Item Identification: **Information technology equipment (EMC directive)**
Universal 48-pindrive programmer

LabProg-48LV

Item Description: 12VAC
16W

The equipment complies with the principal protection requirements of the EMC Directive (Directive 89/336/EEC relating to electromagnetic compatibility) based on a voluntary test.

This attestation applies only to the particular sample of the product and its technical documentation provided for testing and certification. The detailed test results and all standards used as well as the operation mode are listed in

Test report no. E 015L KO

Test standards: EN 55022/1994, EN 55022/A1:1995, EN 55022/A2:1997, EN 55024/1998

This attestation is released with the above mentioned attestation number by the Certification Body of TÜV PRODUCT SERVICE. It does not permit the use of a TÜV PRODUCT SERVICE certification mark.

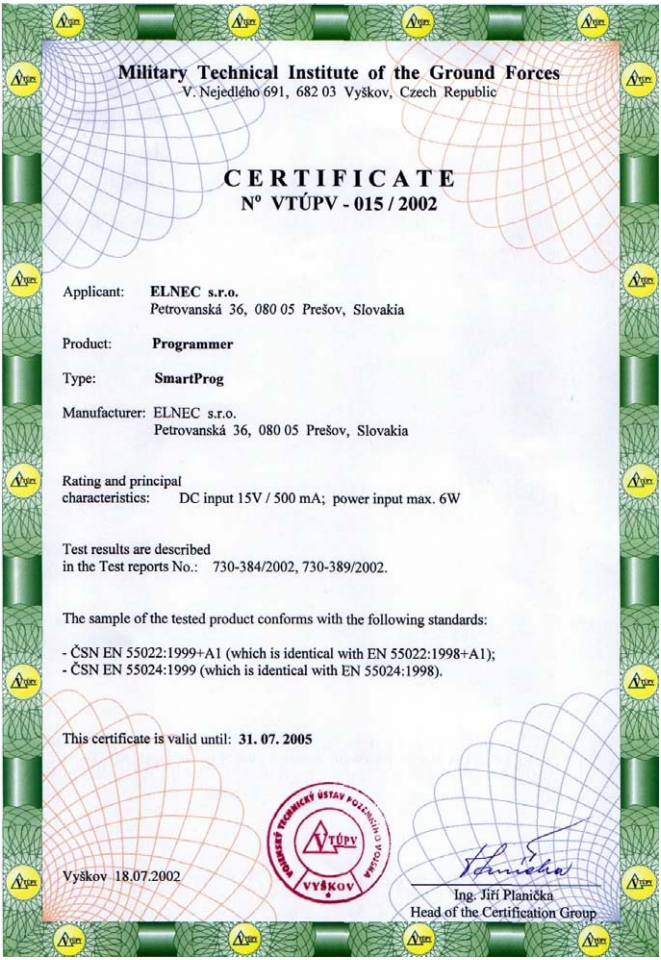
After preparation of the necessary technical documentation as well as the conformity declaration the CE marking as shown below can be affixed on the equipment as stipulated in Article 10.1 of the Directive. Other relevant Directives have to be observed.

CE

Department: EMC / STK / KO
Date: July 22, 1999

TÜV PRODUCT SERVICE GMBH · Zertifizierstelle · Ridlerstrasse 31 · D-80339 München

E/3 – CE Zertifikat SmartProg



Military Technical Institute of the Ground Forces
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
N° VTÚPV - 015 / 2002

Applicant: **ELNEC s.r.o.**
Petrovanská 36, 080 05 Prešov, Slovakia

Product: **Programmer**

Type: **SmartProg**

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 15V / 500 mA; power input max. 6W

Test results are described in the Test reports No.: 730-384/2002, 730-389/2002.

The sample of the tested product conforms with the following standards:

- ČSN EN 55022:1999+A1 (which is identical with EN 55022:1998+A1);
- ČSN EN 55024:1999 (which is identical with EN 55024:1998).

This certificate is valid until: **31. 07. 2005**

Vyškov 18.07.2002

VOJENSKÝ TECHNICKÝ ÚSTAV POZEMNÍHO VOJSKA VTÚPV VYŠKOV

Ing. Jiří Planička
Head of the Certification Group

E/4 – CE Zertifikat PREPROM-02aLV

ZERTIFIKAT ◆ CERTIFICATE ◆ 認証書 ◆ СЕРТИФИКАТ ◆ CERTIFICADO ◆ CERTIFICAT

EC Attestation of Conformity

No.: E8 99 07 36878 001

TÜV PRODUCT SERVICE

ELNEC, spol. s.r.o.
Petrovanska 36, Posta 5, P.O.Box 22

SK – 080 05 PRESOV

Item Identification: **Information technology equipment (EMC directive)**
Universal EPROM programmer

PREPROM-02aLV

Item Description: 12V DC
6W

The equipment complies with the principal protection requirements of the EMC Directive (Directive 89/336/EEC relating to electromagnetic compatibility) based on a voluntary test.

This attestation applies only to the particular sample of the product and its technical documentation provided for testing and certification. The detailed test results and all standards used as well as the operation mode are listed in

Test report no. E 016L KO

Test standards: EN 55022/1994, EN 55022/A1:1995, EN 55022/A2:1997, EN 55024/1998

This attestation is released with the above mentioned attestation number by the Certification Body of TÜV PRODUCT SERVICE. It does not permit the use of a TÜV PRODUCT SERVICE certification mark.

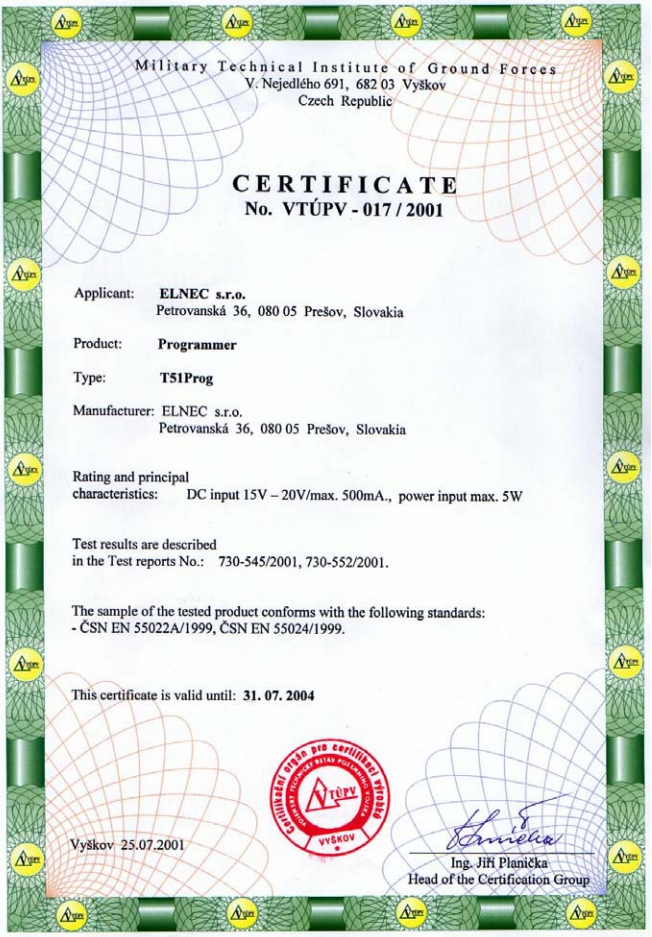
After preparation of the necessary technical documentation as well as the conformity declaration the CE marking as shown below can be affixed on the equipment as stipulated in Article 10.1 of the Directive. Other relevant Directives have to be observed.

CE

Department: EMC / STK / KO
Date: July 22, 1999

TÜV PRODUCT SERVICE GMBH · Zertifizierstelle · Ridlerstrasse 31 · D-80339 München

E/5 – CE Zertifikat T51prog



Military Technical Institute of Ground Forces
V. Nejedlého 691, 682 03 Vyškov
Czech Republic

CERTIFICATE
No. VTÚPV - 017 / 2001

Applicant: **ELNEC s.r.o.**
Petrovanská 36, 080 05 Prešov, Slovakia

Product: **Programmer**

Type: **T51Prog**

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 15V – 20V/max. 500mA., power input max. 5W

Test results are described in the Test reports No.: 730-545/2001, 730-552/2001.

The sample of the tested product conforms with the following standards:
- ČSN EN 55022A/1999, ČSN EN 55024/1999.

This certificate is valid until: **31. 07. 2004**

Vyškov 25.07.2001

Ing. Jiří Planička
Head of the Certification Group

E/6 – CE Zertifikat 51&AVRprog

Military Technical Institute of Ground Forces
V. Nejedlého 691, 682 03 Vyškov
Czech Republic

CERTIFICATE
No. VTÚPV - 009 / 2000

Applicant: **ELNEC s.r.o.**
Lomnická 2, 080 05 Prešov, Slovakia

Product: **Programmer**

Type: **51&AVRprog**

Manufacturer: ELNEC s.r.o.
Lomnická 2, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 12 V/0,5 A; power input 3 W.

Test results are described at the Test Reports No.: 730-401/2000, 730-407/2000.

The sample of the tested product conforms with the following standards:
- ČSN EN 55022B/1996, ČSN EN 61000-4-2/1997, ČSN EN 61000-4-3/1997.

This certificate is valid until: **30.06.2003**

Vyškov 06.06.2000

Ing. Jiří Planička
Head of the Certification Group

E/7 – CE Zertifikat PIKprog+

Military Technical Institute of the Ground Forces
V. Nejedlého 691, 682 03 Vyškov, Czech Republic

CERTIFICATE
N° VTÚPV - 016 / 2002

Applicant: **ELNEC s.r.o.**
Petrovanská 36, 080 05 Prešov, Slovakia

Product: **Programmer**

Type: **PIKprog +**

Manufacturer: ELNEC s.r.o.
Petrovanská 36, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 12V / 500 mA; power input max. 5W

Test results are described in the Test reports No.: 730-385/2002, 730-388/2002.

The sample of the tested product conforms with the following standards:

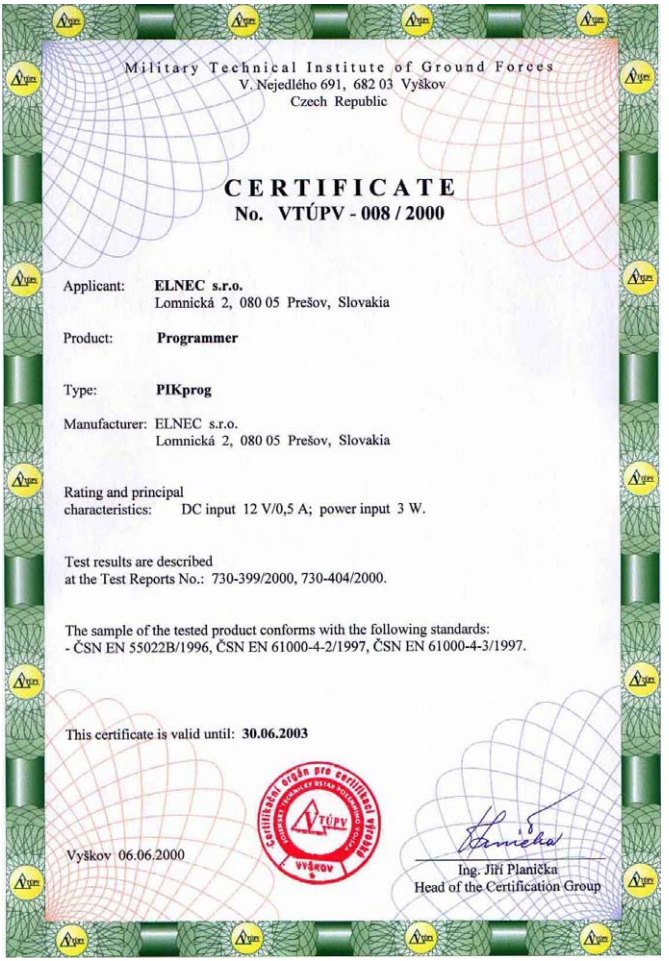
- ČSN EN 55022:1999+A1 (which is identical with EN 55022:1998+A1);
- ČSN EN 55024:1999 (which is identical with EN 55024:1998).

This certificate is valid until: **31. 07. 2005**

Vyškov 18.07.2002

Ing. Jiří Planička
Head of the Certification Group

E/8 – CE Zertifikat PIKprog



Military Technical Institute of Ground Forces
V. Nejedlého 691, 682 03 Vyškov
Czech Republic

CERTIFICATE
No. VTÚPV - 008 / 2000

Applicant: **ELNEC s.r.o.**
Lomnická 2, 080 05 Prešov, Slovakia

Product: **Programmer**

Type: **PIKprog**

Manufacturer: ELNEC s.r.o.
Lomnická 2, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 12 V/0,5 A; power input 3 W.

Test results are described at the Test Reports No.: 730-399/2000, 730-404/2000.

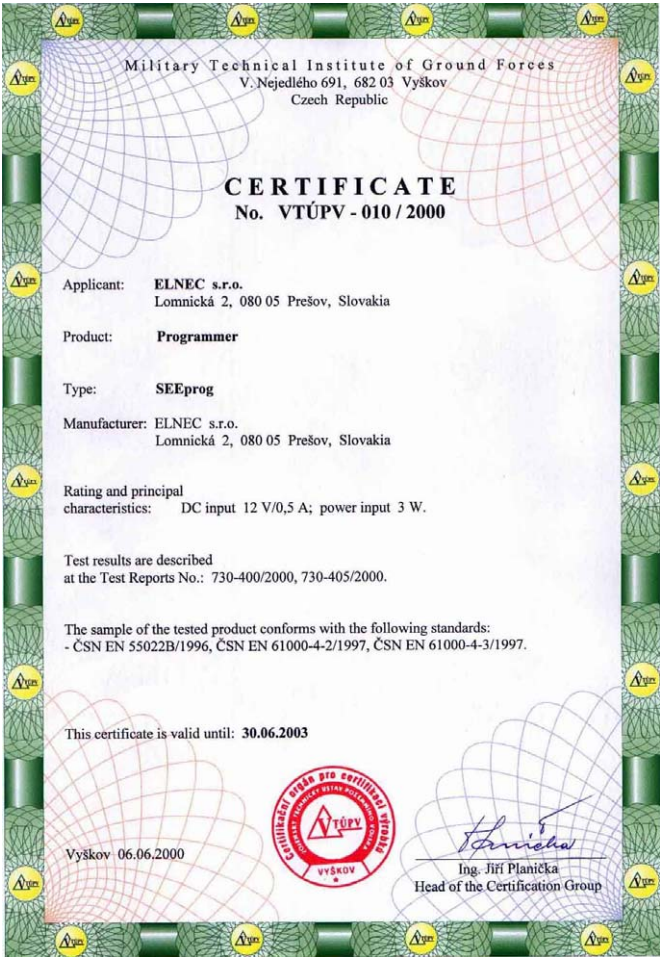
The sample of the tested product conforms with the following standards:
- ČSN EN 55022B/1996, ČSN EN 61000-4-2/1997, ČSN EN 61000-4-3/1997.

This certificate is valid until: **30.06.2003**

Vyškov 06.06.2000

Ing. Jiří Planička
Head of the Certification Group

E/9 – CE Zertifikat SEEprog



Military Technical Institute of Ground Forces
V. Nejedlého 691, 682 03 Vyškov
Czech Republic

CERTIFICATE
No. VTÚPV - 010 / 2000

Applicant: **ELNEC s.r.o.**
Lomnická 2, 080 05 Prešov, Slovakia

Product: **Programmer**

Type: **SEEprog**

Manufacturer: ELNEC s.r.o.
Lomnická 2, 080 05 Prešov, Slovakia

Rating and principal characteristics: DC input 12 V/0,5 A; power input 3 W.

Test results are described at the Test Reports No.: 730-400/2000, 730-405/2000.

The sample of the tested product conforms with the following standards:
- ČSN EN 55022B/1996, ČSN EN 61000-4-2/1997, ČSN EN 61000-4-3/1997.

This certificate is valid until: **30.06.2003**

Vyškov 06.06.2000

Ing. Jiří Planička
Head of the Certification Group